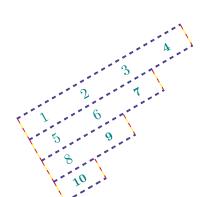
Tabu Package Updates

January 12, 2019

The tabu package patches internal commands of many packages and is liable to stop working if those internal interfaces change.

Unfortunately the original author appears to be out of contact and the package had not been updated for some years.

This manual documents version 2.8 from 2011, but the distributed package contains fixes reported since that time, see https://github.com/tabu-fixed/tabu

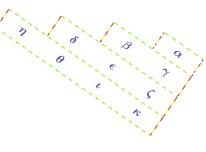




tabu and longtabu

Flexible Let tabulars

2011/02/26 – version 2.8 release



Abstract

This package defines a single environment tabu to make all kinds of tabulars in text or in math mode provided that they do not split across pages.

An environment longtabu – based on D. Carlisle longtable package – is also provided to make tabulars that can stretch out on several pages, while keeping some features (not all of them) of the tabu environment.

tabu is more flexible that tabular, tabular*, tabularx and array and extends the possibilities. All tabulars in this document were made with the tabu environment, of course... The implementation is optimised to minimise the measurements required to put all together.

 $\mathcal{T}_{\aleph}b\subset$ likes colors too, with special lines that are able to keep the alignment of the surrounded text... and also like numbers with the possibility to embed siunitx S (or s) columns. $\mathcal{T}_{\aleph}b\subset$ does not modify any of the macro defined by array.sty or in the \LaTeX kernel¹.

 $\mathcal{T}_{\aleph}b \subset \text{requires } \varepsilon\text{-TEX}$ and the standard package array.sty. Natural widths of columns are computed (but not printed) by the code of varwidth by D. Arseneau. Finally longtabu is based on longtable.

Contents

Sı	ummary of the features provided by $\mathcal{T}_{\aleph}b$ \subset	3
1	Examples and counterexamples	5
	1.1 "Locally global" settings and their scopes	
	1.2 X column widths computation	
	1.3 Inserting Verbatim material (fancyvrb)	7
	1.4 Maths inside tabu X columns	7
	1.5 Embedding sunitx S columns inside X columns	8
2	The tabu environment	8
	2.1 tabu, tabu to and tabu spread	8
	2.2 longtabu, longtabu to and longtabu spread	9
	2.3 tabu X columns – Mastering horizontal space	10
	2.3.1 X columns with "tabu spread"	
	2.3.2 Negativ width coefficients for X columns	
	2.3.3 Multicolumn in tabu	
	2.4 \tabulinesep and \extrarowsep - Mastering vertical space	
	2.5 tabu in math mode	14
3	Lines leaders and colors inside tabu	14
	3.1 First important remark	14
	3.2 Vertical lines: has an optional parameter	14
	3.3 Multiple \firsthline and \lasthline	15
	3.4 More style for lines	16
	3.5 Automatic horizontal lines and row colors	17
4	Modifying the font and the alignment in one row: \rowfont	18
5	Saving and restoring a tabu: \savetabu, \usetabu and \preamble	19

This documentation is produced with the DocStrip utility, and required $T_{\aleph}b \subset$ with its linegoal option.

[→] To get the package, To get the documentation

run: run (thrice):

etex tabu.dtx pdflatex tabu.dtx

To get the index,

run:

makeindex -s gind.ist tabu.idx The .dtx file is embedded into this pdf file thank to embedfile by H. Oberdiek.

6	Some other features	
	6.1 Printing numbers inside tabu with numprint and siunitx	. 20
	6.1.1 \tabudecimal	
	6.1.2 You should know how it works	0.1
	6.2 Paragraph indentation	
	6.3 delarray shortcuts	
7	Differences between tabu, tabular, tabularx and longtable	
	7.1 Paragraph indentation	
	7.2 Custom environments	
	7.3 Inversion of tokens	
	7.4 Improved process for rewriting columns (for keen readers)	
8	The package options	
	8.1 The debugshow package option	
	8.2 The delarray package option	
	8.3 The linegoal package option	. 24
9	Corrections of some bugs (available only inside tabu)	. 24
	9.1 Correction for colortbl and arydshln: compatibility with delarray	. 24
	9.2 Correction for arydshln: @ columns	. 25
10	To do for even better tabus	. 25
	TECHNICAL NOTICE AND IMPLEMENTATION	
11	11.1 Drawing a tabular - The $\tau_8 b \subset$ approach	
	11.2 Algorithms	
	11.3 The tabu strategies	
	11.4 Identification and Requirements	
	11.5 Flow chart of expansion	
	11.7 Rules, colors and vertical adjustment	
	11.8 The entry inside tabu	
	11.9 The rewriting process: inside the "\@mkpream group"	
	11.10 Implementing the strategy at the exit of the \@mkpream group	
	11.11 One trial after the other (\tabuestrategy) 11.12 The algorithms: Measuring the tabu box	
	11.13 Measuring the natural width of columns (varwidth code from D. Arseneau)	
	11.14 Measuring the height and depths of rows	
	11.15 \tabuphantomline	. 71
	11.16 Horizontal lines inside tabu: \tabucline, \firsthline and \lasthline	
	11.18 Verbatim inside tabu with X columns.	
	11.19 \savetabu	
	11.20 \rowfont	
	11.21 Taking care of footnotes and \arraybackslash	
	11.22 Corrections 11.23 Package options and Initialisation	
10	<u> </u>	
12	References	91
13	History	
	[2011/02/26 v2.8]	
	[2011/02/25 v2.7]	
	[2011/02/19 v2.5]	
	[2011/02/17 v2.4]	. 91
	[2011/02/13 v2.3]	
	[2011/02/12 v2.2 – New implementation - Absolutely no modification of array.sty]	
	[2011/01/19 v2.1] [2011/01/18 v2.0]	
	[2011/01/15 v1.9]	
	[2010/12/28 v1.8]	
	[2010/12/18 v1.7]	
	[2010/12/07 v1.5]	
	[2010/11/12 v1.4] [2010/11/18 v1.3]	
	[2010/11/15 v1.2]	
	[2010/10/28 v1.1]	. 94
14	Index	94

C
W
4
- /
-8
100
(
- 5
- (
-
6
6
- 5
- 0
(
(
_
loo loo
- 2
0
C
200
- 0
- 6
_
-
·
7
Н
Ų,

Summary of	the features provided by $\tau_{\aleph}b$
tabu	is like tabular in text mode and like array in math mode when
	there is no X column in its preamble
longtabu	is like longtable with the possibility to use tabu X columns and
	vertical lines with the extended syntax.
[tabu] to $\langle extit{dimen} angle$	specifies the target width of the whole tabular. This is like
	tabular* with an automatic stretchability that can be overwritten
	with @{\extracolsep {dimen}} in front of the preamble.
$\{ exttt{tabu}\}$ $ exttt{spread}$ $ exttt{}$ $ exttt{}$ $ exttt{}$ $ exttt{}$ $ exttt{}$	has no equivalent in \LaTeX : the final width is $\langle dimen \rangle$ wider than
	the natural width that can be obtained with spread Opt.
[width,color]	vertical lines have an optional parameter.
X[coef,align,type]	X columns widths are adjusted in order for the whole tabular to
X[coef,align,type,\$]	fit the target width. The target width is a dimension either: → directly specified with {tabu} to⟨dimen⟩
	→ computed from the natural width: {tabu} spread⟨dimen⟩
	→ by default \linewidth (or \linegoal with the linegoal package
	option). coef scales the widths of the X columns, if there are more than
	one X column.
	align is either r, c, 1 or j (or R C L J) and type can be p
	(default), m or b.
	$X[\$]$ makes a math X column (ie. $\{\$\}X < \{\$\}$)
	X[\$\$] display math X column: >{\$\displaystyle }X<{\$}
X[-coef,align,type]	X columns widths are first computed with the absolute value
	coef . Then the width is made narrower down to the natural width of the column if possible.
	In any case, the final width does not exceed the one obtained with X[coef].
X[X options]{S[S options]}	Embed a siunitx S column into a tabu-X column.
\everyrow {code}	Allows to add horizontal lines automatically for every row.
(everylow (code)	The settings can be changed inside the tabu
\rowfont [align]{font spec}	Modify the font and optionally the alignment of each cell in one
\\\	row.
$ ag{dimen}$	More control on vertical spacing of lines in a way very close to
\	cellspace's method (dynamic vertical spacing adjustment).
$ackslash ext{extrarowsep} = raket{ ext{dimen}}$	Control vertical spacing (\extrarowheight and
	\extrarowdepth): fixed vertical spacing adjustment
	\tabulinesep generally gives better results.
\tabudecimal {\usermacro }	a help to align numbers easily inside a column.
\savetabu {user-name}	Saves the tabu preamble and its parameters. The command must
\usetabu {user-name}	appear at the end of a line. Makes a tabu of exactly the same shape as the one saved with
	\savetabu. All parameters (target, preamble, stretch etc.) are restored.
	This command is put alone in the preamble in place of the
	columns specifications.
\preamble {user-name}	Makes a tabu with the same preamble as the one saved with
· · · · · · · · · · · · · · · · · · ·	\savetabu. The only preamble is restored, not the target nor
	any other parameter.
	This command is put alone in the preamble in place of the

~
11
4
- /
- 5
- (
7
- (
- 3
- (
,
٠,
- 1
- (
- 2
- (
- 2
6
0.000
[See 1 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
- (
- 7
- 3
- (
- 7
- '
_
- (
- 6
_
വ
•
_
C
-
- >
- 0
- 3
- 7
_
J
٠,
_0
7
ŀ
_
•

Summary of the	features provided by $ au_{\aleph}b$
\tabulinestyle {line spec}	Sets the current line style to be used for and \tabucline
\newtabulinestyle {name=spec,}	Defines a line style for use with \tabucline [name] or with Iname
\tabucline [spec]{start-stop}	Draws a line comparable to \hline. The line \langle spec \rangle can contain information for making a dash or dotted line (f.ex. [on 3pt off 6pt]) and a color name. The line spec can also be defined with \newtabulinestyle
\taburulecolor dbl rule sep {rule color}	sets the color for rules (\hline, \firsthline)
$\verb \taburowcolors [skip] \\ \langle number \rangle \{first last\}$	Sets the color series to make alternate background colors for rows
\tabuphantomline	inserts a phantom (<i>ie.</i> invisible) line inside the tabu May be usefull with \multicolumn in some cases.
$\tracingtabu = 0, 1, 2, 3, 4$	Reports informations in the .log file about the steps of the algorithm for tabu X columns, and the informations saved by \savetabu.

5 / 101

$\mathsf{T}_{\aleph}b \subset [\text{rev.}2.8 \text{ release}] \otimes 2010 - 2011 \Leftrightarrow \mathsf{FC}$

1 Examples and counterexamples

Let's begin in colors!

 $\mathsf{T}_{\aleph}b \subset$ provides facilities to put horizontal and vertical leaders in a tabular. The package xcolor must be loaded of course. Background colors for cells are left to package colortbl which is fully compatible with $\mathsf{T}_{\aleph}b \subset$.

1.1 "Locally global" settings and their scopes

\tabulinestyle
\taburulecolor
\taburowcolors
\everyrow

 $\mathsf{T}_{\aleph}b$ cobserves $\mathsf{T}_{\mathsf{E}}\mathsf{X}$ grouping levels for the settings of rule colors (\tabuluecolor) and styles (\tabuluecolor), and \everyrow. There is however a subtility for nested tabu environments as described in this example:

Listing 1: Locally global settings and their scopes

Color of the TEX group

Color of the last end-of-line setting

Color of the TEX group

>	Here the lines	are drawn in blue
	But starting from here	they are green coloured!
	And now a nested tabu	guess what colour
		is used for rules?

Inside the group, rule colors are blue. After the group, rule colors are red again!

The "rules" are the following:

- If outside of a tabu environment, the settings are local to the TEX group. Every tabular drawn inside this group will inherit from the settings of that group.
- If \taburulecolor (or \everyrow or \tabulinestyle) is used inside a cell of the tabular, this is the same: the settings a local to that cell, and any nested tabular will inherit from the setting of that cell.
- When used after the end of a row, the settings are globally changed from that point until the end of the tabular, or until a new setting is set at the end of a further row (TeXnically, this is done inside a \noalign group). But a nested tabu does not inherit from this "global" setting, and inherits from the settings of the TeX group instead.

If \arrayrulecolor or \doublerulesepcolor (from package colortbl) are used instead of \taburulecolor then colors are globally overwritten.

A counterexample from the xcolor package: \rowcolors does not like \cline, \cmidrule etc.²

```
row 1
test
            row 2
test
           row 3
test
            row 4
test
\operatorname{test}
           row 5
       other row 6
test
       other row 8
test
       other row 9
test
```

The \rownum counter is not reliable in the case of \cline or \cmidrule.

In addition, the first coloured row is yellow, while one could have expected it green...

For $\mathcal{T}_{\aleph}b \subset \text{color changes}$ are called at \everyrow:

test	row 1
test	row 2
test	row 3
test	row 4
test	row 5
test	other row 6
test	other row 7
test	other row 8

 $\mathcal{T}_{\aleph}b \subset$ does not use "real" alternate colors but colorseries provided by package xcolor. This allow some gradations:

```
Row number 1=1
test
     Row number 2=2
test
test
     Row number 3=3
test
     Row number 4=4
     Row number 5=5
test
     Row number 6=6
test
     Row number 7=7
test
     Row number 8=8
test
     Row number 9=9
test
     Row number 10=10
test
```

1.2 X column widths computation

The new algorithm implemented in version 2.8 requires only one measure of the width of the table in any case. This speeds up the convergence of the algorithm.

```
      \begin{tabu} to 140mm {|X[1,1] | X[2,c] | X[3,c] | X[1,r]|}

      |\dotfill | & Text & Text & Text \

      Text & Text & Text & Text

      \end{tabu}

      Text | Text | Text | Text |
```

$$X = (140mm - 8 \times \text{tabcolsep} - 5 \times \text{varrayrulewidth}) / 7 = 17.4896mm$$

^{2.} Because color changes are done at \everycr, which is not exactly the same as $T_{\aleph}b \subset \text{\text{everyrow}}!$

$\mathsf{T}_{\aleph}b \subset [\text{rev.}2.8 \text{ release}] \otimes 2010 - 2011 \Leftrightarrow \mathsf{FC}$

1.3 Inserting Verbatim material (fancyvrb)

Though the content of the tabu environment is collected for measuring purpose, it is possible to insert verbatim material with the tabu* variant of the environment. The content is then carefully collected and re-scanned (with \scantokens). During the process, the @ letter is read with the category code it has been given at the entry inside the environment (it is possible to say \makeatletter before \begin {tabu*}).

Example:

Ιt is possible to insert Verbamaterial with some \csname control sequences \endcsname inside a tabu and inside X columns. Negativ coefficients work well too, adjusting the width of the X column to the natural width if it is finally less than the width computed with the absolute value of the coefficient.

A complete Verbatim environment is also admissible.

But you must use the star form of the environment: tabu* which uses \scantokens.

Verbatim environments must be put alone on their lines (in the input file) for nothing is allowed after \begin{Verbatim} or \end{Verbatim}.

Another point to know is that \begin and \end control sequences should match otherwise, you must enclose the Verbatim environment inside braces.

This is related to the fact that tabu collects its body, and looks for matching pairs of \begin ... \end !

tabu* is useless when nested inside another tabular. The star form of the environment should be used only for the outermost table! Comments are removed, unless the % character is given a category code of 12 (or 11) before the entry inside the environment.

```
\tabulinestyle{on2pt Crimson!60 off3pt yellow!50} \tabulinesep=2mm
\makeatletter \@makeother\%
\begin{tabu*}spread Opt {|X[-1]X|} \tabucline -
This is a small \Verb+\Verbatim+\par
insertion
&
\begin{Verbatim}{[listparameters={\topsep=-\ht\strutbox}]}
And this is a complete % with some comments
Verbatim environment % every now and then
\end{Verbatim}
\\ \tabucline -
\end{tabu*}
```

Here a small **\Verbatim** insertion

And this is a complete % with some comments Verbatim environment % every now and then

It's not possible to insert a lstlisting environment presently, but you can save such an environment in a \vbox and insert it inside the tabu of course.

1.4 Maths inside tabu X columns

X[\$\$] columns

$\mathsf{T}_{\aleph}b \subset [\text{rev.2.8 release}] \otimes 2010 - 2011 \Leftrightarrow \mathsf{FC}$

1.5 Embedding sunitx S columns inside X columns

A S column from siunitx can be embedded into a X column of $\tau_{\aleph}b$... with the following limitations:

- The X column must be centered: X[c] to keep the alignment,
- The optional alignment parameter of \rowfont must not be used.

```
\newcolumntype Y{S[group-four-digits=true,
                   round-mode=places,
                   round - precision =2,
                   round - integer - to - decimal=true,
                   per-mode=symbol,
                   detect - all]}
\tabucolumn Y
\tabulinestyle {1pt GreenYellow}
\begin{tabu}spread 8pt{|*2{Y|}c} \tabucline -
\rowfont\bfseries
      {January} &{February} &... \\
                \tabucline[1pt on2pt GreenYellow
                   1 -
               &745.32
      12.324
                         &... \\
      21.13 &0
                         &... \\
      213.3245 &12.342
                         &... \\
      2143.12 &324.325 &... \ tabucline -
\end{tabu}
```

January	February	•••
12.32	745.32	
21.13	0.00	
213.32	12.34	
2143.12	324.33	

Column widths are not exactly the same

```
\newcolumntype Z{X[c]{%
   S[group-four-digits=true,
      round-mode=places,
      round-precision=2,
     round - integer - to - decimal=true,
      per-mode=symbol]}}
\tabucolumn Z
\begin{tabu}spread 8pt{|*2{Z|}c} \tabucline -
\rowfont\bfseries
      {January} &{February} &... \\
        \tabucline[1pt on2pt GreenYellow]-
      12.324 &745.32 &... \\
                         &... \\
              &0
      213.3245 &12.342 &... \\
      2143.12 &324.325 &... \\\tabucline -
\end{tabu}
```

January	February	
12.32	745.32	
21.13	0.00	
213.32	12.34	
2143.12	324.33	

Column widths are exactly the same

\tabucolumn is there to say $\tau_{\aleph}b \subset$ that the column type has to be treated with a high priority in the rewriting process.

Another possibility to print number is provided with \tabudecimal.

2 The tabu environment

2.1 tabu, tabu to and tabu spread

```
\begin {tabu} [pos] {tabular preamble} \begin {tabu} to \langle dimen \rangle [pos] {tabular preamble} \begin {tabu} spread \langle dimen \rangle [pos] {tabular preamble}
```

The tabu environment behaves mostly like tabular: the preamble is parsed by the macros in array.sty and some measures are performed before printing. tabu improves tabular and array:

- footnotes and index words are allowed inside tabu, unlike tabularx, footnote links are not broken when compiled with hyperref. The syntax \footnote [number]{\langle text\rangle} is allowed in tabu and longtabu (this is not implemented for longtable yet...)
- X columns are implemented with an *optional* parameter for the **width-coefficient** (which can be negativ: see next section), the **alignment** (r, c, l, or j, and R, C, L or J for

 $\mathsf{T}_{\aleph}b \subset [\text{rev.}2.8 \text{ release}] \otimes 2010 - 2011 \Leftrightarrow \mathsf{FC}$

ragged2e settings) and the **column type** (p, m, or b). tabu has a default target width when used with X columns, making nesting even easier.

- You are used to the tabular environment in text mode, and array environment in math mode, but tabu works in both modes and its name does not change... X columns are also possible in math mode; delarray shortcuts for delimiters are available in both math and text modes.
- A tabu environment can contain another tabular of any kind: tabular, tabular*, tabular* or tabu itself can be placed in any cell of a tabu. Conversely, tabu can be placed in a tabular, tabularx etc..
- tabu provides facilities for vertical and horizontal lines, and for the insertion of verbatim text inside X columns.
- tabu is fully compatible with colortbl, delarray, hhline, makecell, booktabs, siunitx, dcolumn, warpcol, etc.. When you are inside a tabu environment, you can use \raggedleft, \raggedright and \centering without special care about \arraybackslash and conversely \\ has its "normal" meaning inside a list of items that may appear in a X column...

\begin {tabu} to\(\dimen\) is like tabular* but the inter-columns space is given a stretchability of 1fil, in other words $\$ {\extracolsep {0pt plus 1fil}} is inserted by default at the beginning of the tabular preamble, unless another value for \extracolsep is specified. Therefore "tabu to" fills in width the specified $\langle dimen \rangle$.

\begin {tabu} spread $\langle dimen \rangle$ does a tabular whose width is $\langle dimen \rangle$ wider than its natural width. $\mathbb{Q}\{\text{extracolsep \{0pt plus 1fil}\}\}\$ is inserted by default if $\langle dimen \rangle > 0$.

2.2 longtabu, longtabu to and longtabu spread

```
\begin {longtabu} [l | c | r] {tabular preamble} \begin {longtabu} to \langle dimen \rangle [l | c | r] {tabular preamble} \begin {longtabu} spread \langle dimen \rangle [l | c | r] {tabular preamble}
```

longtabu is just like tabu but page breaks are allowed between rows of the table. longtabu is based on the longtable package which must be loaded, and all features of the longtable environment works inside longtabu: \endhead, \endfirsthead, \endfoot, \endlastfoot and \caption.

longtabu enhances the longtable environment with the possibility to use X columns and line specifications for horizontal and vertical rules. longtabu is thus much easier than ltxtable.

The following commands provided for tabu do not work with longtabu:

tabu command	Not available	Not (yet) implemented	Comment
\tabucline		*	\tabucline does not care of page breaks presently: use \hline instead.
\usetabu	×		but \savetabu and \preamble work.
mathematical mode	×		longtable is not designed to work in math mode.
delarray shortcuts	×		a delimiter cannot be spanned over
			pages
\tabuphantomline	×		useless inside longtabu

However, tabu X columns, \rowfont, \extrarowsep, \tabulinesep, \tabudecimal, \tabucline (with restrictions on page breaks), \taburulecolor, \tabulinestyle, \taburowcolors, \preamble, \langtabu\} to , \langtabu\} spread work inside longtabu.

$\mathsf{T}_{\aleph}b$ c [rev.2.8 release] @2010-2011 $ext{c}$

2.3 tabu X columns – Mastering horizontal space

tabu X columns can be viewed as an enhancement of tabularx X columns, but do not interact with them, for they are defined only for a short time during the parsing of the preamble:

• width coefficients can optionally be given to X columns ex. X[2.5]X[1] is the same as X[2.5]X and the same as X[5]X[2] This means that the first X column will be two and a half wider than the second one or that the first X column width will be 5/7 of the whole tabular width.

X[2.5]	X

• negativ width coefficients can be given to X columns:

ex. X[-2.5]X[1] or X[-2.5]X or X[-5]X[2]

In this case, the first X column will be $at\ most$ two and a half wider than the second one, and if the $natural\ width$ of the first X column is finally less than $2.5\times$ (the width of the second column) then it will be narrowed down to this natural width.

The following tabus have the same preamble:

\begin ${\text{tabu}}$ to \linewidth ${|X[-2.5c]|X[c]|}$:

X[-2.5]	X	
Negativ o	coefficients make X columns close to standard	X
	1, c and r columns.	

• horizontal alignment specification is made easier with X[5,r]X[2,c] for example. Vertical alignment can be specified as well with X[5,r,m]X[2,p,c] (commas are not required, but X[2cm] or X[4pc] could be misunderstood – not by TeX: by you...).

Modifier	Meaning	Default
l,c,r,j,L,C,R,J	left, centered, right, justified	j
p, m, b	X column is converted into p, m or b column	р
\$	$X[\$]$ is a shortcut for: $>\{\$\}X<\{\$\}$	
\$\$	X[\$\$] is a shortcut for: >{\$\displaystyle }X<{\$}	

- tabu X columns can be spanned with \multicolum.
- tabu X columns can be used with "tabu spread" for small tabulars.
- tabu X columns can contain any type of tabular, tabular*, tabularx or tabu without special care about the syntax. tabu can also be put inside tabular, tabular* and tabularx. As long as tabu with X columns has a *default target*, nesting tabu with X columns is easy. Furthermore, the default global alignment of a nested tabu is t (for top) while the default global alignment of a tabu in a paragraph is c (for centered).
- The "algorithm" (or the arithmetic) to get the target width for tabu X columns is the same as the one used by tabularx. \hfuzz is the "tolerance" for the whole tabular width. We use \$\varepsilon\$-TFX \dimexpr

instead of T_EX primitives (with round/truncate bias correction).

- Convergence to the target width is optimised: the \halign preamble is not re-built at each trial, but only expanded again, until the target is reached. Though optimized, the process is the same as the one implemented for tabularx and in particular the content of the tabu environment is collected as soon as a tabu X column is found in the preamble. This implies restrictions on catcode modifications and verbatim text inside a tabu with X columns.
- If the width of the whole tabular is not specified with "tabu to" it is considered to be \linewidth. The linegoal package option makes the default width equal to \linegoal. Compilation must then be done with pdfTEX either in pdf or dvi mode, and package linegoal is loaded. \linegoal requires pdfTEX for its \pdfsavepos primitive and the zref-savepos: if the tabu is not alone in its paragraph ie.if the target is not \linewidth, then two compilations (or more) are required to get the correct target.

 Default target for nested tabu environments is always \linewidth, which equals to the
 - Default target for nested tabu environments is always \linewidth, which equals to the column width inside p, m, b and X columns.
- As long as the \halign content is expanded more than once, protections against counters

incrementation, whatsits (write) index entries, footnotes etc.. are set up: the mechanism of tabularx is reimplemented and enhanced for tabu X columns. \tabuDisableCommands can be used to neutralize the expansion of additional macros during the trials.

X columns with "tabu spread"

tabu X columns can be used with "tabu spread" to adjust the column widths of tabulars that contain only small pieces of text. The question is: how to make a tabular the width of the line, with 6 columns; the columns 1, 2, 5 and 6 are of equal widths and the widths of columns 3 and 4 are only one half. As possible solution:

 $\begin{array}{ll} \begin{array}{ll} \begin{array}{ll} \begin{array}{ll} \begin{array}{ll} \begin{array}{ll} \end{array} & \end{array} & \begin{array}{ll} \end{array} & \begin{array}{ll} \end{array} & \begin{array}{ll} \end{array} & \begin{array}{ll} \end{array} & \end{array} & \end{array} & \begin{array}{ll} \end{array} & \end{array} & \begin{array}{ll} \end{array} & \end{array} & \end{array} & \begin{array}{ll} \end{array} & \end{array} & \begin{array}{ll} \end{array} & \end{array} & \begin{array}{ll} \end{array} & \end{array} & \begin{array}{ll} \end{array} & \end{array} & \begin{array}{ll} \end{array} & \end{array} & \end{array} & \begin{array}{ll} \end{array} & \end{array} & \end{array} & \begin{array}{ll} \end{array} & \end{array} & \begin{array}{ll} \end{array} & \end{array} & \begin{array}{ll} \end{array} & \end{array} & \end{array} & \begin{array}{ll} \end{array} & \end{array} & \end{array} & \begin{array}{ll} \end{array} & \end{array} & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \begin{array}{ll} \end{array} & \end{array} & \end{array} & \begin{array}{ll} \end{array} & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \begin{array}{ll} \end{array} & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \end{array} & \\ & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \begin{array}{ll} \\ & \end{array} & \end{array} & \begin{array}{ll$ 1 & 2 & 3 & 4 & 5 & 6 \\hline \end{tabu}

1	2	3	4	5	6

But the text in each cell is very short: one single character, and you prefer the table to be tight, but don't know the exact width of the whole:

1 & 2 & 3 & 4 & 5 & 6 \\\hline \end{tabu}

|--|

But now it's definitely too narrow, then give it some more space:

1 & 2 & 3 & 4 & 5 & 6 \\\hline \end{tabu}

1	2	3	4	5	6

tabu spread is useless with long columns: the following tabular was made with this preamble:

"Like the air we breathe, Sherlock Holmes is everywhere. His pipesmoking, deer stalkered image peers at us from ads in Yellow Pages, to signs for neighbourhood crime-watch; from billboards to the classroom; from film and television to the public library, and now over the Internet. He long ago transcended the boundaries of 19th Century London³ to become an international best-seller and has been accepted as part of British folklore. Holmes is alive to millions."

There the text was too long, and tabu spread behaves as if you didn't give it a target.

The result of this example is the same as if one had written \begin {tabu}to\linewidth.

Sherlock Holmes

The "official" web site: http://www.sherlockholmes.com/

In the preamble, $\mathbb{Q}\{\}$ means that the margin is removed.

Negativ width coefficients for X columns

```
\tabulinestyle {3pt ForestGreen}
\tabucline - \savetabu{FirstNegativTest}
  \beta \left( X[-1]X[-1]c] \right)
    \alpha & \beta \\
    \gamma & \delta + \epsilon + \zeta + \eta + \theta
 \end{tabu}$
 This is a tabu with negativ width coefficients for \text X columns
 \\ \tabucline -
\end{tabu}
```

$$\begin{pmatrix} \alpha & \beta \\ \gamma & \delta + \epsilon + \zeta + \eta + \theta \end{pmatrix}$$
 This is a tabu with negative width coefficients for X columns

 $\left(\begin{array}{cc}
\alpha & \beta \\
\gamma & \delta + \epsilon + \zeta + \eta + \theta
\end{array}\right)$

And this is the same with \t abulinesep set to 2pt.

Multicolumn in tabu

\tabuphantomline

The process of \multicolumn implies the TeX primitive \omit which discards the tabular preamble for the spanned columns. Discarding the preamble means discarding the information about the widths of the columns. This explains why the following example does not work properly:

```
\begin{tabu}{|X|X|X[2]|} \tabucline-\\ multicolumn2{|c|}{Hello} & World \\ \tabucline-\\ end{tabu}
```

The correct result can be obtained by the mean of a phantom line, that will remain invisible unless your preamble contains special **©** or ! columns that prints some text:

```
\begin{tabu}{|X|X|X[2]|} \tabucline-\\ \multicolumn2{|c|}{Hello} & World \ \tabucline-\\ \tabuphantomline \\ \end{tabu}
```

Hello	World
-------	-------

Remember you may need \tabuphantomline in conjunction with \savetabu and \usetabu with \multicolumn. Even if it is possible to add a \tabuphantomline in any line of the tabu, it is a good practice to append it at the end of the tabu, for it may introduce indesirable side effects on vertical alignment otherwise, when tabu is nested inside another tabular.

In particular, \tabuphantomline should not be followed by \cr or \\ or \tabularnewline...

The need for this command could disappear in a future release, but this requires a complete new implementation of \multicolumn...

2.4 \tabulinesep and \extrarowsep - Mastering vertical space

```
\label{eq:continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous_continuous
```

\tabulinesep sets the minimal vertical space allowed between the cell content and the cell border. The macro may be prefixed by \global (even inside a \noalign group)⁴.

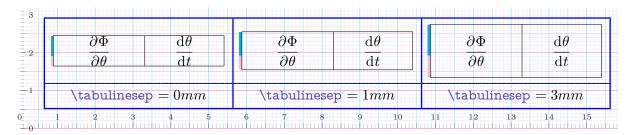
It is possible to set the "top limit" (a TEX dimension called **\abovetabulinesep**) and the "bottom limit" independently with the syntaxes:

```
\label{linesep} $$ \begin{array}{ll} \text{ sets Abovetabulinesep} \\ \text{ tabulinesep} =_{\langle dimen \rangle} & \text{ sets Abovetabulinesep} \\ \text{ tabulinesep} =_{\langle dimen \rangle}^{\langle dimen \rangle} & \text{ sets Abovetabulinesep} \\ \text{ sets Abovetabulinesep} \\ \text{ sets Abovetabulinesep} \\ \text{ sets Abo
```

These parameters can be used in text and math modes to give more vertical space between lines, especially when using math formulae.

Examples (with $\tracingtabu = 3$ and interfaces- $\protect\$ papergraduate to see the struts):

^{4.} However \tabulinesep is not a dimension! You can't test, for example, \text{ifdim \tabulinesep} > 0pt! Test \abovetabulinesep and \belowtabulinesep instead, if needed.



\tabulinesep is a soft parameter, and leads to rows which do not share the same height.

```
 \begin{array}{l} \langle \text{extrarowsep} = \langle \text{dimen} \rangle \\ \langle \text{extrarowsep} = ^{\langle} \text{dimen} \rangle \\ \langle \text{extrarowsep} = _{\langle} \text{dimen} \rangle \\ \langle \text{extrarowsep} = _{\langle} \text{dimen} \rangle ^{\langle} \text{dimen} \rangle \\ \langle \text{extrarowsep} = _{\langle} \text{dimen} \rangle ^{\langle} \text{dimen} \rangle \\ \end{array}
```

\extrarowsep is an extra vertical space which is added to each row, inconditionally. array.sty provides the TeX dimension \extrarowheight and $\mathcal{T}_{\aleph}b \subset$ provides \extrarowdepth in addition.

As a result, the rows can share the same height/depth but the spacing is not dynamic. \tabulinesep can be used even with positive values for \extrarowsep, for tabu inserts only one strut per row and vertical spacing computations are possible in all cases.

The macro can be prefixed by \global as well, even inside a $\noalign group^5$.

Set \extrarowheight and \extrarowdepth to different values, with the syntaxes:

```
\label{eq:continuous} $\operatorname{extrarowsep} = \widehat{\langle dimen \rangle}$ sets $\operatorname{extrarowheight}$ $\operatorname{extrarowdepth}$ is unchanged $\operatorname{extrarowsep} = _{\operatorname{dimen}}^{\operatorname{dimen}}$ sets $\operatorname{extrarowdepth}$ is unchanged $\operatorname{extrarowsep} = _{\operatorname{dimen}}^{\operatorname{dimen}}$ sets $\operatorname{extrarowdepth}$ and $\operatorname{extrarowheight}$.
```

Both \extrarowheight and \extrarowdepth are scaled by \arraystretch (a scaling $macro^6$ of array.sty) if \arraystretch > 1...

These parameters can be used in text and math modes.

Examples (with $\tracingtabu = 3$ and interfaces- $\protect\$ papergraduate to see the struts):

Standard c column	X[-1,c] columns	Math mode	Mixed: X[-1c]X[-1c\$]
One Two	One Two	αβ	First $\frac{\partial \Phi}{\partial \theta}$
Three Four	Three Four	$rac{\Phi}{ heta}$ Γ^t_x	Second $\frac{\mathrm{d}\theta}{\mathrm{d}t}$
	\extraro	wsep $=3mm$	
1 2 3	\extraro		11 12 13 14 15
1 2 3 Standard c column	\extraro 4 5 6 7 X[-1,c] columns		11 12 13 14 15 Mixed: X[-1c]X[-1c\$]
Standard c column One Two	4 5 6 7 X[-1,c] columns One Two	Math mode $\frac{\alpha \beta}{\beta}$	Mixed: X[-1c]X[-1c\$]
Standard c column	4 5 6 7 X[-1,c] columns	8 9 10 Math mode	Mixed: X[-1c]X[-1c\$]

^{5.} However \extrarowsep is not a dimension! You can't test, for example, \ifdim \extrarowsep > 0pt! Test \extrarowheight and \extrarowdepth instead, if needed.

^{6. \}arraystretch is not a dimension but a macro that stores a scaling factor.

2.5 tabu in math mode

On the left, you can see the famous Maxwell-Lorentz equations for electromagnetic field in vacuum, publicated in 1873.

In this example, the big tabu is: \begin {tabu} to\linewidth {XX[-1\$]}.

The nested tabu (in math mode) uses delarray shortcut: its preamble is: \begin{tabu}({rl}.

\tabulinesep has been set to 2pt. Horizontal rules are booktabs \toprule and \bottomrule.

array	tabu	tabu spread $1em$
$\left egin{array}{cc} lpha & eta \ \gamma & \delta \end{array} \right $	$\left egin{array}{cc} lpha & eta \ \gamma & \delta \end{array} \right $	$\left egin{array}{cc} lpha & eta \ \gamma & \delta \end{array} \right $

Here, vertical lines are made with delarray shortcuts:

\$\begin{tabu} spread 1em |{cc}|

Vertical lines inside the tabular preamble gives:

$$\begin{array}{c|c} \alpha & \beta \\ \gamma & \delta \end{array}$$

This was an example of \savetabu...\usetabu to keep the alignment.

3 Lines leaders and colors inside tabu

3.1 First important remark

The features provided in this section are quite experimental: they are not generally taken for good typography. You can use $\mathcal{T}_{\aleph}b \subset$ with package booktabs for example, which provides properly designed commands for horizontal rules in tabulars. arydshln is pretty good too, but it modifies a huge amount of macros of array.sty, something that $\mathcal{T}_{\aleph}b \subset$ does not.

Lines in tabu printed in this document are mostly made with booktabs.

3.2 Vertical lines: | has an optional parameter

Inside tabu environment, the vertical line marker | has an *optional* argument which is the width of the vertical rule. The default width remains \arrayrulewidth of course. The optional argument for | can also contain the name of a color. color *names* are only possible, not a color specification by the mean of a color model. The width of the line if specified, must come before the color name and... as for X columns parameters, commas are optional.

Example:

This example was printed inside a tabu* whose preamble is: X[-1m] X[m] X[-2m]

It is not a necessary to protect the optional argument with braces: $[\{...\}]$. because $\mathcal{T}_{\aleph}b \subset$ takes care the | token to be rewritten before any other column type (the same for tabu X columns, and siunitx S columns). The rewriting process is divided into three stages under control inside a tabu environment.

$\mathsf{T}_{\aleph}b \subset [\text{rev.2.8 release}] \otimes 2010 - 2011 \hookrightarrow \mathsf{FC}$

3.3 Multiple \firsthline and \lasthline

\firsthline [extratabsurround] make multiple lines! \firstline [extratabsurround]\hline \lastline [extratabsurround] \lastline [extratabsurround]\hline

\firsthline and \lasthline are defined in array.sty and can be used to preserve the alignment of text, when using horizontal lines. Besides, the optional argument can be used to change (locally) the \extratabsurround dimension.

The example of array documentation is:

Tables	with no	versus	Tables	with no	versus
	line			line	
	commands			commands	
	used			used	
tables	with some	used.	tables	with some	$\frac{1}{2}$ used.
	line			line	
	commands			commands	
with \firsthline and			with \hline		
		(text alignment is not			
	\lasthline			preserved)	

Now with tabu you can make double, triple (or more) \firsthline or \lasthline as in:

$^{Align}_{Snment}$	Tables \begin \tabu\[t]\{c\} \with no\\\ line \\\ commands \\\\ used \end \tabu\} \versus tables \begin \tabu\[t]\{ c \} \firsthline \hline \hline \with some \\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\	Tables tables	with no line commands used with some line commands	versus used.
B_{Ottom}^{Oottom}	Tables \begin \tabu\[t]\{c\} with no\\ line \\ commands \\ used \end \tabu\} versus tables \begin \tabu\[b]\{ c \} \firsthline \hline \hline with some \\ line \\ commands \\ \lasthline \hline \hline \hline \end \tabu\} used.	Tables tables	with no line commands used with some line commands	versus

\firsthline \firsthline \firsthline

is equivalent to: \firsthline \hline \hline \alphaline \hline \hl

But the optional argument must come in first position: \firsthline [extratabsurround] ...

The same for \lastline.

In yellow you can see the \extratabsurround strut, because \tracingtabu = 3 for this tabu

$\mathcal{T}_{\aleph}b$ c [rev.2.8 release] © 2010 – 2011 c9 F6

3.4 More style for lines

```
\label{eq:linear_color} $$ \aligned & \ali
```

\taburulecolor sets (in a "locally-global" way) the color to be used for \hline, \firsthline, \lasthline and also vertical lines if the standard line style is used (the standard line style is active after \tabulinestyle {} or after \tabureset).

The optional parameter enclosed by vertical bars: $|\langle double\ rule\ sep\ color \rangle|$ is the color to set between two adjacents rules. If not specified, double (or triple...) rules are separated by a vertical space (\vskip).



And the next paragraph follows...

\tabulinestyle $\{\langle line\ style\ specification \rangle\}$

\tabulinestyle sets the line style for vertical (|) and horizontal lines (ie.\tabucline: \hline, \firsthline etc. are not modified by \tabulinestyle)

The line specification is of the form:

```
3pt rule color on 4pt dash color off 5pt gap color
rule color on 4pt dash color off 5pt gap color
on 4pt dash color off 6pt gap color
3pt rule color
on 4pt dash color
off 5pt
3pt
Named style defined by \newtabulinestyle
```

Well... any parameter is optional. Obviouly the rule color is the same as the dash color and the former overwrites the latter if both are given.

Your color names can contain spaces but:

- If the first character in the line specification is not a letter, then it is taken as a dimension: the thickness of the line. Otherwise, the default thikness is used *ie*.\arrayrulewidth.
- Your color names must not contain any series of characters that match one the patterns: on?

where ? is a character of category 12, different from ! and possibly preceded by spaces. I don't think this is a real limitation...

```
\label{line spec.} $$\operatorname{\colored} $$\operatorname{\colo
```

This command defines a line style to be used in the first optional argument of \tabucline (horizal lines) or the optional argument of | (vertical lines) or with \tabulinestyle (locally-global style).

Style names and color names are babel-protected.

```
\tabucline [style or spec.]{start-end}
```

\tabucline is an attempt to give a versatile command to make horizontal lines:

• \tabucline is pretty good with vertical lines even if the thickness of the line grows up,

[1pt on 1.5pt off 2pt]

[1.5pt]

default

[on 2pt red]

- \tabucline takes care of \extrarowheight,
- \tabucline can make horizontal dashed lines, with a pgf/TikZ syntax: \tabucline [\langle width \rangle on \langle dash \rangle off \langle gap \rangle] \{ \langle first column \rangle \langle last column \rangle \}
- alternatively, you can give \tabucline a \hbox to make a leader with it: The \(spec. \) must then begin with \hbox, \box or \copy,
- finally you can give \tabucline a color name, after the line specification.

Any parameter can be omitted.

\tabucline [1pt on 1.5pt off 2pt]{1-4}	draws a horizontal dashed line of width 1pt. Dashes are 1.5pt long and gap width is 2pt. The line is drawn between columns 1 and 4. Here there are only 2 columns and the line stops at column 2.
\tabucline [1.5pt]{-}	draws a horizontal solid line of width 1.5pt between the first and the last column.
\tabucline {2-}	draws a horizontal solid line of width \arrayrulewidth between the second column and the last one.
\tabucline [on 2pt red]{-5}	draws a horizontal dashed line between columns 1 and 5 of width \arrayrulewidth. Dashed are 2pt long and gap width is 4pt (the default).

Define the line style

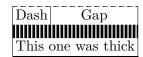
Use the line style

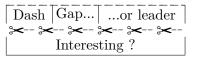
\newtabulinestyle \{myline=0.4pt on 2.5pt off 1pt red\} \tabucline \[myline]\{-\}

Or use a leader or a box to make a leader with it directly in the argument of \tabucline

\tabucline [\hbox {\$\scriptstyle \star \$}]{1-3}

Dashed or dotted
And below is the default





3.5 Automatic horizontal lines and row colors

\everyrow {code}

\everyrow can be used to insert horizontal lines automatically:

This is	a small example	of a tabu
which	automatically	inserts
a horizontal	line after	each of its
		row

\everyrow can be used in longtabu as well. The syntax is like \everycr: a token-like syntax, and braces are mandatory: \everyrow {argument}.

\taburowcolors [first line] $\langle number \rangle$ {first .. last}

\taburowcolors sets the alternate colors to be used on every row of the tabular. The command can be used before a tabu environment or inside it, at the end of a row.

The optional parameter [first line] tells the first row from which background colors are starting – this optional parameter has no effect when \taburowcolors is used at the end of a row: background are starting immediately in this case.

 $\langle number \rangle$ is the number of colors in the color series. If not specified, it defaults to 2 (for alternate rows color).

Finally $\langle first \rangle$ and $\langle last \rangle$ are the first and the last colors in the colorseries.

Example:

```
\taburowcolors[2] 3{Crimson!30 ...
                          ForestGreen!40}
\taburulecolor | GreenYellow | {OrangeRed}
\arrayrulewidth=1pt \doublerulesep=1.5pt
\everyrow {\hline\hline}
\begin{tabu} \{X[-1]X\} \
This is
          &just a test
and i think &it will
look
            &rather bad
            &i've not
for
chosen
           &the colors
with care. &i can't
            &less...
\taburowcolors 2{Crimson .. ForestGreen}
            &This is Crimson
2
            &This is ForestGreen
3
            &This is Crimson
            &This is ForestGreen \\
\end{tabu}
```

This is	just a test
and i think	it will
look	rather bad
for	i've not
chosen	the colors
with care.	i can't
say	less
1	This is Crimson
2	This is ForestGreen
3	This is Crimson
4	This is ForestGreen

\tabureset

To go back to "standard" parameters, $\mathcal{T}_{\aleph}b\subset$ provides the command \tabureset which basically does:

4 Modifying the font and the alignment in one row: \rowfont

\rowfont [alignment] { font specification }

Inside a tabu environment, you can modify the font for each cell in a row. \rowfont has priority over column font specification, exactly like \rowcolor (package colortbl) has priority over \columncolor.

The alignment of each cell in one row can also be changed to:

```
l = left or for ragged2e settings: L c = center C r = right R j = justify J
```

Any other value for the optional $\langle alignment \rangle$ parameter is silently ignored. If ragged2e is not loaded, L R C and J are synonymous with the lowercase equivalent.

```
\label{likelihood} $$ \operatorname{tabu}_{|X|X[-1]|} \to \operatorname{tabucline}_{\operatorname{confont}[c]\setminus bfseries} $$ This & Sis & \operatorname{tabucline}_{\operatorname{confont}[c]} \to \operatorname{tabucline}_{\operatorname{confont}[c]} $$ \operatorname{confont}_{\mathbb{C}} \to \operatorname{tabucline}_{\operatorname{confont}[c]} $$ \operatorname{confont}_{\mathbb{C}} \to \operatorname{confont}_{\mathbb{C}} $$ is $$ is $$ \operatorname{confont}_{\mathbb{C}} \to \operatorname{confont}_{\mathbb{C}} $$ is $$
```

This	Is
tabu	package
for	tabu and longtabu

5 Saving and restoring a tabu

$\strut_{\alpha} (user-name)$

The command \savetabu can be used at the end of any line of a tabu environment to save the parameters of a tabu environment. The saving is always global. This allows to easily make tabulars which share exactly the same shape throughout your document. This can also be used as a kind of tabbing environment which is able to remember the tabs positions...

If the $\langle user-name \rangle$ has been used before, an info is displayed in the .log file and the previous settings are overwritten.

With the \tracingtabu > 0, informations about the saved parameters are reported in the .log file.

Recalling saved parameters are done with \usetabu (complete recovery) or \preamble (partial recovery of the preamble only).

$\usetabu \{\langle user-name \rangle\}$

\usetabu is the complement of \savetabu: it can be put alone in the tabu preamble instead of the usual columns specifications to restore any previous settings saved with \savetabu.

The $\langle user-name \rangle$ must exist otherwise, you get an error.

\usetabu is a help to make several tabulars of exactly the same shape, same target, same preamble. The only parameter that can be changed is the optional vertical position parameter for the whole tabular.

\usetabu does not work with longtabu.

\usetabu locally restores:

- the preamble⁷.
- the vertical position [c], [b] or [t], unless another position is specified.
- the target width of the tabu in points: the saved target width does not contain any control sequence: it is fixed and stored in points.
- the width of tabu X columns: those widths are not calculated any more even in the case of negativ coefficients and X columns are directly transformed into p, m or b columns of the same widths as the ones that where calculated at the time of \savetabu
- \tabcolsep (or \arraycolsep in math mode) \extrarowheight, \extrarowdepth, \arraystretch and \extratabsurround
- \arrayrulewidth, \doublerulesep and the parameters for \everyrow \taburulecolor, \tabulinestyle, and \taburowcolors
- \minrowclearance, (package colortbl)

\abovetabulinesep and \belowtabulinesep are not restored, because they are related to the content of the tabular rather than to its shape.

Example:

```
\tabcolsep=12pt \extrarowsep=1mm
\tabulinestyle {on 1pt ForestGreen}
```

^{7.} The complete \halign-preamble is restored.

<pre>\begin{tabu} to .7\linewidth{ XXX X[c] } \savetabu{mytabu} \tabucline - This & is & tabu & package \\ \tabucline - \end{tabu}</pre>
This is tabu package
\tabureset
\begin{tabu}{\usetabu{mytabu}} \tabucline -
\multicolumn3{ c}{This is tabu} & package \\ \tabucline -
\tabuphantomline
\end{tabu}
F1
This is tabu package

If one day you use tabu, you will have the idea to restore a tabu while modifying its target, or adding new columns... \savetabu and \usetabu have not been thought for this purpose, and you may have unexpected results.

\preamble $\{\langle user-name \rangle\}$

\preamble can also be used after \savetabu. This is a variant of \usetabu that locally restores:

- the tabu (or longtabu) preamble.
- the vertical position [c], [b] or [t] (or [c], [l] or [r] for longtabu), unless another position is specified.
- the tabu / longtabu target width, unless another target is specified.

Any other tabular parameter is not restored.

Put $\prescript{preamble {\langle user-name \rangle}}$ alone inside the tabu (or longtabu) preamble in place of the usual columns specifications.

\preamble works exactly as if you defined a custom environment for tabu.

\preamble works with longtabu .

Example (continued...):

```
\tabulinestyle{1pt off1pt}
\begin{tabu} to\linewidth{\preamble{mytabu}} \tabucline -
This &is &tabu &package \\ \tabucline -
\end{tabu}
```

			,
m:-:-	<u>.</u> _	4 - 1	1 1
i i nis	18	tabu	ı backage i
i			I

\tabcolsep, rule colors etc. are not restored from \savetabu: the only tabu preamble is restored.

6 Some other features

6.1 Printing numbers inside tabu with numprint and siunitx

\tabudecimal

 $\mathsf{T}_{\aleph}b \subset \mathsf{provides}$ a facility to print numbers inside columns. This facility is not implemented to replace siunitx S and s columns or numprint n and N columns or other packages that provide alignment such as warpcol, dcolumn or rccol. It just make easy to apply a macro you get already on each number in a column of a tabu.

\tabudecimal has been developed mainly because it makes possible to align numbers inside tabu X columns.

$\mathsf{T}_{\aleph}b$ c [rev.2.8 release] $@2010 - 2011 ext{c}$

\tabudecimal $\{\langle user-macro \rangle\}$

\tabudecimal can be used in the preamble of a tabu before a column specification. The $\langle user-macro \rangle$ is a macro with one parameter that has to be defined before.

Example with \numprint:

\rowfont [c]{\bf } January & February \\
12.324 & 745.32 \\
21.13 & 0 \\
213.3245 & 12.342 \\
2143.12 & 324.325 \\
\end {tabu}

January	February	•••
12,32 €	745,32 €	
21,13 €	0,00 €	
213,32 €	12,34 €	
2 143,12 €	324,33 €	

Example with \SI:

\def\usermacro#1{\SI[group-four-digits=true, % thousand separator round-mode=places, % round numbers round-precision=2, % with 2 decimal digits round-integer-to-decimal=true, % add trailing 0 if necessary per-mode=symbol]{#1}{\officialeuro\per\kilo\gram}}

January	February	•••
12.32€/kg	745.32€/kg	
21.13€/kg	0.00€/kg	
213.32€/kg	12.34€/kg	
2143.12€/kg	324.33€/kg	

As you can see, the columns widths are exactly the same, whatever their content.

Here \tabulinesep has been set to 3pt.

You should know how it works...

Yes you should know how it works to avoid problems. tabu has a small scanner based on $\$ turrelet to grab all numbers, blank spaces, commas and dots + and - sign and also the letter e and E for exponents. The scanner stops as soon as something else than a number, blank space, comma, dot, +, -, e, E is found, and even if it is a macro that contains a number.

This explains why there is \zap@space in the definition of \usermacro: because the scanner scans blank spaces and because \numprint does not allow blank spaces in its mandatory argument, quite strangely...

6.2 Paragraph indentation

tabu takes care of paragraph indentation when it is used with X columns and its default target, no matter if it has been loaded or not with the linegoal option. Example with LATEX default: \parindent = 20pt.

This is tabu with its default target in an indented paragraph.

This is tabu with its default target, preceded by \noindent

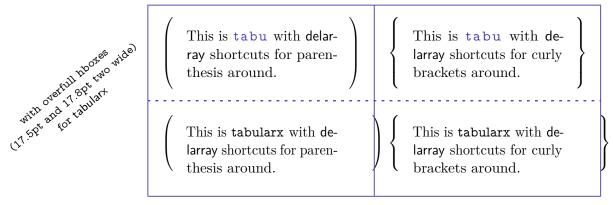
This is tabularx with target: \linewidth in an indented paragraph.

This is tabularx with target: \linewidth, preceded by \noindent

$\mathsf{T}_{\aleph}b = [\text{rev.2.8 release}] \otimes 2010 - 2011 \Leftrightarrow \mathsf{FC}$

6.3 delarray shortcuts

When you enclose your tabular with math delimiters using delarray shortcuts, $\tau_{\aleph}b \subset$ tries to reach its target for the whole: the tabular and the delimiter(s). You can see the difference:



Here \t tabulinesep = 3mm

7 Differences between tabu, tabular, tabularx and longtable

7.1 Paragraph indentation

See Paragraph indentation

7.2 Custom environments

Unlike tabularx, it is possible to define your own environment using tabu:

\newenvironment{foo}
 {\begin{tabu}{X[1.2]|[1pt gray]X}}
 {\end{tabu}}

tabu environment, even when X columns are used, may appear in the definition of your custom tabular environment.

You can also use the commands \savetabu \preamble (or \usetabu) for this purpose.

7.3 Inversion of tokens

When you typeset the following tabular:

\begin{tabular}{|>{\bfseries}>{ before }|<{ one }<{ two }|}
 cell content
\end{tabular}</pre>

You get the following result: before **cell content two one**

 \longrightarrow The word *before* is not bold, and **two** comes before **one**.

The reason is explained in the documentation of array.sty, and is related to the array environment in math mode when using \newcolumntype.

This rather strange inversion of tokens may be justified in math mode (otherwise, errors may occur) but not in text mode in our opinion. Inside a tabu environment, when not in math mode, the tokens are not reversed and you get the intuitively expected result:

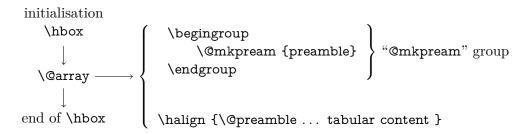
before cell content one two

In math mode however, tokens are in the reverse order in the tabu environment like they are in the array environment.

$\mathsf{T}_{\aleph}b \subset [\text{rev.}2.8 \text{ release}] \otimes 2010 - 2011 \Leftrightarrow \mathsf{FC}$

7.4 Improved process for rewriting columns (for keen readers)

Any tabular that does not split across pages is made with the following process:



For more details, see the Flow chart of expansion.

\@mkpream works in two times inside a (semi-simple) group:

First the rewriting process:

Each special column in the tabular preamble is transformed into one the columns defined by array.sty.

Second the building of the \halign preamble:

The "rewritten preamble" is parsed and transformed in a preamble for the TeX primitive \halign. The result is stored into the \@preamble macro.

Any special columns of tabu are defined only inside the "@mkpream" group.

In the following example, you get an error with tabular and no error with tabu. With tabular, and siunitx S column, the *rewriting process* is as follow:

\documentclass {minimal}
\usepackage {numprint,siunitx,xcolor}
\usepackage {tabu}
\begin {document}

\begin {tabular}{*2{S[color=green]}}
123,45
\end {tabular}

\begin {tabu}{*2{S[color=green]}}
123,45
\end {tabu}

\end {document}

Inside tabular:

- 1) Rewrite S: not found because inside {...}
- 2) Rewrite *
- 3) Rewrite ${\tt n}$ column defined by package numprint

Then the 'n' in green is rewritten \longrightarrow problem

Inside tabu:

- 1) Rewrite *
- 2) Rewrite | (there is none here)

go back

- 3) Rewrite *
- 4) Rewrite |
- 5) Rewrite S
- 6) Rewrite $n \longrightarrow not$ found because S was rewritten before, according to siunitx definition.

The process of rewriting columns is usually longer inside tabu than inside tabular, but conversely tabu with X columns is optimised compared to tabularx, because the preamble is built only once, and not rebuilt before each trial as tabularx does. Thus tabu is much quicker than tabularx.

The process of rewriting is very sensitiv to the order in which columns are actually rewritten. This becomes critical when columns are defined with an optional argument like tabu X and I columns or siunitx S column.

23 / 101

$\mathsf{T}_{\aleph}b \subset [\text{rev.2.8 release}] \otimes 2010 - 2011 \Leftrightarrow \mathsf{FC}$

8 The package options

8.1 The debugshow package option

```
\tracingtabu \tracingtabu = 1, 2, 3 or 4
```

The control sequence \tracingtabu has the same effect as the debugshow option:

- $\mathsf{T}_{\aleph}b\subset$ will report the widths it computes at each attempt to read the target, when X columns are used.
- Saved informations on the tabu are reported in the .log file when \savetabu is used.

```
\tracingtabu = 2 gives more information on the measures of the natural widths.
```

\tracingtabu = 3 shows the struts inserted inside the tabu environment and gives more information about the measures of the height and depth of every row.

\tracingtabu = 4 displays information on the insertions made by \tabucline.

Typical information in the .log file:

```
(tabu) Try
            tabu X
                       tabu Width
                                                 Coefs
                                                          Update
                                      Target
(tabu)
      1)
          386.67296pt
                       797.34592pt
                                    386.67296pt
                                                 2.0pt
                                                        -205.33649pt
(tabu)
      2)
          181.33647pt
                       386.67294pt
                                    386.67296pt
                                                 2.0pt
                                                         0.00002pt
(tabu)
```

What does it mean?

- 1) The first attempt was performed with X=386.67296pt The tabu width (797.34592pt) exceeded the target by 410.67296pt. Thus X has been updated: 410.67296pt / 2 = 205.33649pt and then: X = 386.67296pt 205.33649pt = 181.33647pt
- 2) The second attempt lead to a tabu width of 386.67294pt: the target is reached. The final width of each X column is the product of tabu X by its width coefficient.

8.2 The delarray package option

delarray option has the single effect to load delarray.sty for delimiters shortcuts around tabu. Delimiters shortcuts work both in math and text mode.

8.3 The linegoal package option

With the linegoal option, the default target for tabu with X columns is \linegoal instead of \linewidth. The linegoal package must be loaded and compilation must be done with pdfTEX, otherwise, a warning is displayed and the linegoal option has no effect: the default target remains \linewidth. \linegoal works with pdfTEX in pdf mode and in dvi mode.

If for some reason, you wish to turn down the linegoal option in your document, you can say (in a group for example): \let\tabudefaulttarget=\linewidth

In any case, specifying the target overwrites the default: \begin {tabu} to\linewidth

9 Corrections of some bugs (available only inside tabu)

9.1 Correction for colortbl and arydshln: compatibility with delarray

Both colortbl and arydshln forget the control sequence \@arrayright in their implementation, quite strangely because both of them take care of \@arrayleft. As a result, delarray shortcuts for delimiters around a tabular does not work if colortbl and/or arydshln are loaded.

Those control sequences are used by the delarray package to put variable size delimiters around

the array:		
$\begin{array}{c} \begin{array}{c} \\ \\ \end{array} \end{array}$		$\ \ \ \ \ \ \ \ \ \ \ \ \ $
 \end {tabu}	is like:	 \end {tabu} \right .

9.2 Correction for arydshln: @ columns

A bug in \adl@xarraydashrule: !-arg columns (class 1) and @-arg columns (class 5) should be treated the same as far as rules are concerned.

With this correction, the "known problem number 1" in arydshln documentation is solved.

10 To do for even better tabus

In decreasing order of priority:

- → Make double \tabucline compatible with colortbl \doublerulesepcolor
- → Multiple \tabucline between different columns: extended specs: \tabucline [line spec]{start-stop} start-stop} [line spec]{start-stop} ...
- Reimplement \multicolumn in order to allow the X token in \multicolumn preamble. Provide \multicell to allow spanning columns and rows at the same time.
- → Presently, longtabu with X columns works only if \LTchunksize is greater than the number of rows. I compiled a longtabu of 56 pages on my PC with \LTchunksize = 2000 without problem. Presently \LTchunksize is set to 10000 during trials when longtabu contains X columns.
- → Make \tabucline work with page breaks (one line on the top of the page, one line on the bottom of the previous).

11 Technical notice and Implementation

11.1 Drawing a tabular - The $\tau_{\aleph}b$ = approach

 $\mathsf{T}_{\aleph}b \subset \mathsf{has}$ a different approach than almost any other package providing facilities for tabulars. colortbl and arydshln both put the cells contents into a box for measuring purpose, and then use the dimensions of each box to make their setups:

colortbl needs the dimensions of the box to put a rule in the background of the cell,

arydshin needs the dimensions to set the length of its leaders (dash lines).

This is achieved by modifying the macros defined in array.sty to insert columns inside the \halign preamble.

Instead, $\tau_{\aleph}b \subset$ proceeds as follow:

- It first measures (if there are some negative width coefficients, or if tabu spread is used) the natural widths of the cells / the columns,
- 2. Then it always measures the height and depth of each cell / row,
- 3. Thereafter, the tabular is printed exactly as if array.sty was entitle to print it: no "extra" boxing of the cells material. The measurements have been stored and can be used to set the struts (only one per row) and the lengths of vertical leaders.
- 4. No macros of array.sty is modified at stage 3.

 $\mathsf{T}_{\aleph}b$ material inserted in the tabular for vertical leaders, \rowfont etc. is put inside the special "free" tokens provided by array.sty:

- A vertical leader is put inside a ! column: !{vertical leader}
- Changing font and alignment in one row requires some setup in > tokens: >{rowfont material}.

This way, the commands of array.sty that build each column definition (or preamble, in the sense of halign) are never modified.

11.2 Algorithms

tabu to target

The algorithm of \tabu@arith computes the desired widths to reach the target. In any case, only one measure of the tabular is required to get the widths for all columns. Here we describe the method with an example and some equations too to show that this handle all cases in generality.

Notations and initialisation of X In the case of tabu to the target T = 300 is given: it is the target specified by the user or the default tabu target which is \linewidth $-\langle parindent\ correction \rangle$ or \linegoal. Each X column has a width coefficient which is given too (or default to 1). The coefficients are: $c_1, c_2, \ldots c_n$.

X is the main dimension that drives the widths of all columns with a non negative coefficient, and limit the widths of columns with a negative coefficient.

Then we have first:

Coef c_i		c_1	c_2	c_3	c_4	c_5	c_6	\sum	Δ
		-1	-2	-5	-2	2	3	15	
Target T	300							1	

Some coefficients are negative and we have to measure the natural widths of the corresponding columns, for columns always have a width:

$$\lambda_i = \begin{cases} c_i \cdot X & \text{if } c_i > 0\\ \text{Min}(|c_i| \cdot X, \nu_i) & \text{if } c_i < 0 & \text{with} \quad \nu_i \le T \quad \forall i \end{cases}$$

 ν_i is the "natural width of the column" in the sense that it is the maximum of the natural widths of each cell in the *ith* X column, limited to the tabu target: $\nu_i \leq T \quad \forall i$.

The whole width of the tabular is always:

$$wd(table) = \sum_{i} \lambda_{i} + incompressible material \begin{cases} \bullet \text{ } \text{$tabcolsep}$} \\ \bullet \text{ } vertical lines/leaders thickness} \\ \bullet \text{ } non \text{ X columns} \end{cases}$$

and should finally be equal to T, by the correct computation of the λ_i .

So what is X at first? Columns that have a non negative coefficients always have a width equal to $\lambda_i = c_i \cdot X$ therefore, if we only have non negative coefficients, we can safely set:

$$X = \frac{T}{\sum_{i} c_i}$$

then: $\sum_i \lambda_i = \sum_i c_i \cdot X \geq T$ at the first trial. But this is not the same if some coefficients are negative, because in this case the column width λ_i can shrink until its natural width ν_i and may be until to 0pt! And then if every column has a negative coefficient, one of them can have a width close to the target T. We have to ensure that the first measure of the natural widths does not limit them artificially:

$$\forall i \quad c_i < 0 \implies |c_i| \cdot X \ge T$$

$$\exists c_i > 0 \implies \sum_{\substack{i \\ c_i > 0}} c_i \cdot X \ge T$$

And finally, for the measure: $X = \text{Max}\left[\max_{\substack{i \\ c_i < 0}} \frac{T}{|c_i|}; \frac{T}{\sum\limits_{\substack{i \\ c_i > 0}} c_i}\right]$

Coef c_i		c_1	c_2	c_3	c_4	c_5	c_6	Σ	Δ
		-1	-2	-5	-2	2	3	15	
Target T	300								
X	300							1	
$ u_i$		10	300	80	80				
λ_i		10	300	80	80	600	900	1970	1800

First step of the algorithm: reducing the width After having measured the table we get: wd(table) = 2100. The *incompressible material* is 2100-1970 = 130 wide and the gap to the target is $\Delta = 2100-300 = 1800$.

We now choose a new value for X:

$$\sum_{i} \lambda_{i} = \sum_{i} \min_{\substack{i \\ c_{i} < 0}} (\nu_{i}; c_{i} \cdot X) + \sum_{\substack{i \\ c_{i} > 0}} c_{i} \cdot X \leq \sum_{i} |c_{i}| \cdot X$$

Let's try
$$X' = \frac{\sum_{i} \lambda_i - \Delta}{\sum_{i} |c_i|}$$
 so that $\sum_{i} \lambda'_i \leq \sum_{i} |c_i| \cdot X' \leq \sum_{i} \lambda_i - \Delta$:

Coef c_i		c_1	c_2	c_3	c_4	c_5	c_6	Σ	Δ
		-1	-2	-5	-2	2	3	15	
Target T	300							1	
X	300							1	
$ u_i$		10	300	80	80				
λ_i		10	300	80	80	600	900	1970	1800
X'	11.33								

 $X' = \frac{1970 - 1800}{15} = \frac{170}{15} = 11.33$ Note that the computation of X' does not involve any measurement.

Coef c_i		c_1	c_2	c_3	c_4	c_5	c_6	Σ	Δ
		-1	-2	-5	-2	2	3	15	
Target T	300								
X	300								
$ u_i$		10	300	80	80				
λ_i		10	300	80	80	600	900	1970	1800
X'	11.33								
λ_i'		10,00	22,67	56, 67	22,67	22,67	34,00	168, 67	-1.33

Here we are in the case where the table width:

$$\label{eq:wd} \begin{split} \text{wd(table)} &= \sum_i \lambda_i + \text{incompressible material} = T + \Delta \\ &\implies \sum_i \lambda_i' + \text{incompressible material} \leq \sum_i \lambda_i - \Delta + I = T \end{split}$$

Without any measure, we can say that the final table width will be less than the target, if we choose X'. The free space to share among the X columns (computed with X') is now $\Delta' = T - (\sum_i \lambda_i' + I) = 300 - (168.67 + 130) = -1.33$, where I is the incompressible material.

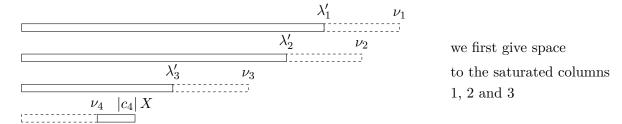
Giving space We say that a column is saturated (ie.full) if its natural width is greater than $|c_i| \cdot X$, or all the same that $\lambda_i < \nu_i$. We also will consider that the columns with $c_i > 0$ have a "natural width" which is always equal to $c_i \cdot X$: in other words, a column with a non negative coefficient is always saturated.

Giving space (or "refunding" space) to the columns must be done in priority to the *saturated* columns. If all columns are finally underfull, then we will distribute the extra space to each, according a distribution rule. But this case can only occur if $\forall i \ c_i < 0$ because we first choosed X so that:

$$X \ge \frac{T}{\sum_{\substack{i \\ c_i > 0}} c_i}$$

and hence, the sum of the widths of the "non negative" columns exceeds the target.

Let's rank the columns widths:



Because of the saturation, the total amount of space to give: $|\Delta|$ shall be shared among the columns according to their widths coefficients. We shall not give too much space: the columns shall remain saturated. Let $0 < \epsilon \le |\Delta|'$ the amount of space to give, then after the operation:

$$\lambda_1$$
" + λ_2 " + λ_3 " = $\lambda_1' + \lambda_2' + \lambda_3' + \epsilon$
= $|c_1|X' + |c_2|X' + |c_3|X' + \epsilon$

Let's say $X'' = X' + \frac{|\Delta|'}{\sum_{\substack{i \ c_i \text{ saturated}}} |c_i|}$ then it's possible, without any measure, to compute:

$$\sum_{\substack{i \\ c_i \text{ saturated}}} \lambda^{"}_i + \nu_4 \le \sum_{\substack{i \\ c_i \text{ saturated}}} |c_i| \cdot X^{"} + \nu_4 \le \sum_{\substack{i \\ c_i \text{ saturated}}} |c_i| X^{\prime} + \Delta^{\prime} = \sum_{\substack{i \\ c_i \text{ saturated}}} \lambda^{\prime}_i + \Delta^{\prime} \le T - I$$

Or for clarity: $\sum_{i} \lambda_{i}$ " + $I = wd(table) \leq T$ and the new free space to share is now:

$$\Delta$$
" = $\left| T - \left(\sum_{i} \lambda$ "_i + I \right) \right|

At each step of the computation, and without any measure but the first, X grows, Δ decreases, and finally the target is reached for X such that $\Delta \leq \text{hfuzz}$.

Coef c_i		c_1	c_2	c_3	c_4	c_5	c_6	Σ	Δ
		-1	-2	-5	-2	2	3	15	
Target T	300								
X	300							1	
$ u_i$		10	300	80	80				
λ_i		10	300	80	80	600	900	1970	1800
X'	11.33							 	
λ_i'		10,00	22,67	56,67	22,67	22,67	34,00	168, 67	-1.33
<i>X</i> "	11.43							 	
λ " $_i$		10,00	22,86	57, 14	22,86	22,86	34, 29	170,00	0

Now if the width of the table is less that the target, because 1) every column has a negative coefficient and 2) their natural widths are so small than the tabular don't fill the wanted horizontal space, the algorithm artificially raise the natural widths, according to a linear distribution:

$$\lambda_i' = \lambda_i + \Delta \cdot \frac{\lambda_i}{\sum_i \lambda_i} = \nu_i + \Delta \cdot \frac{\nu_i}{\sum_i \nu_i} = \nu_i \cdot \left(1 + \frac{\Delta}{\sum_i \nu_i}\right)$$

tabu spread dimen

The case of tabu spread is interesting and quite complex...

Here, the aim of the game is to give a target to the table, depending on its natural width. tabu has a default target (\linewidth in general, but it is possible to \let \tabudefaulttarget to another value... for example \linegoal) which is a maximum for the final target of tabu spread. The case where the spread is 0pt is not simpler nor more difficult.

If every column has a negative coefficient, it's rather easy because either the table exceeds the target, and then the new target will be the default target (the maximum), or the table width is less than the default target and we fix the new target to be that width + the spread, in the limit of the default target.

The condition that must hold on coefficient is not restritive if every column has a negative coefficient because if you say, for example: $X = \underset{i}{\operatorname{Max}} \frac{\nu_{i}}{|c_{i}|}$ then:

$$\sum_{i} \lambda_{i} = \sum_{i} \operatorname{Min} (\nu_{i}; |c_{i}| \cdot X)$$

is true. It's always possible to find a X such that the behaviour anounced in the documentation is observed!

Then let's get some non negative coefficients. The natural widths of such columns must be measured, but the natural width of the tabular is not the same, for the proportions between column widths – expressed by their positive coefficient c_i – must be respected.

The real natural width of the tabular, which observe the proportions between columns with a non negative coefficient is:

$$wd(table) + \underset{c_{i}>0}{\text{Max}} \left(\frac{\nu_{i}}{c_{i}}\right) \times \sum_{\substack{i \ c_{i}>0}} c_{i} - \sum_{\substack{i \ c_{i}>0}} \nu_{i} > wd(table)$$

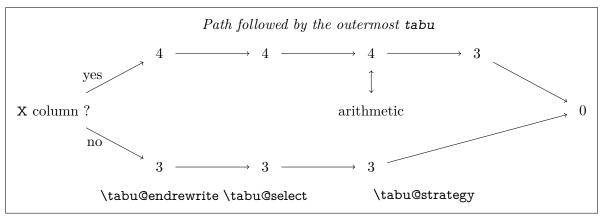
This quantity is computed, $\mathcal{T}_{\aleph}b \subset$ adds the spread and fix the new target to the sum, in the limit of the default target.

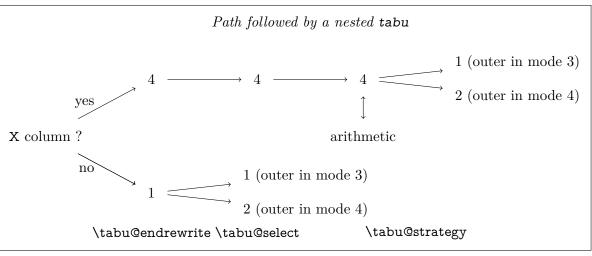
Then
$$X$$
 is initialized such that: $X = \text{Max}\left[\max_{\substack{i \\ c_i < 0}} \frac{T}{|c_i|}; \frac{T}{\sum\limits_{\substack{i \\ c_i > 0}} c_i}\right]$

and the algorithm described in the former section works, without any new measurement of the tabular. Unless this was not possible or deemed inconvenient for clarity, the code is presented in the same order it executes.

11.3 The tabu strategies

		Not nested (outer)	Nested			
	\count@	condition	\count@	condition		
			0	outer is in mode 0		
\tabu@endrewrite	3	no X column	1	no X column		
	4	X columns	3	X column		
			0	outer in mode $0 \Rightarrow \text{print}$		
\tabu@select			1	outer in mode 3		
			2	from 1 in $\t 0$ in mode 4		
	3 or 4	needs trials	3 or 4	needs trials		
	0	print out				
			1	Exit in vertical measure (outer in mode 3)		
\tabu@strategy			2	Exit with a rule (outer in mode 4)		
	3	Vertical measure				
	4	Horizontal measure	4	Horizontal measure (nested in coef< 0 or spread)		





11.4 Identification and Requirements

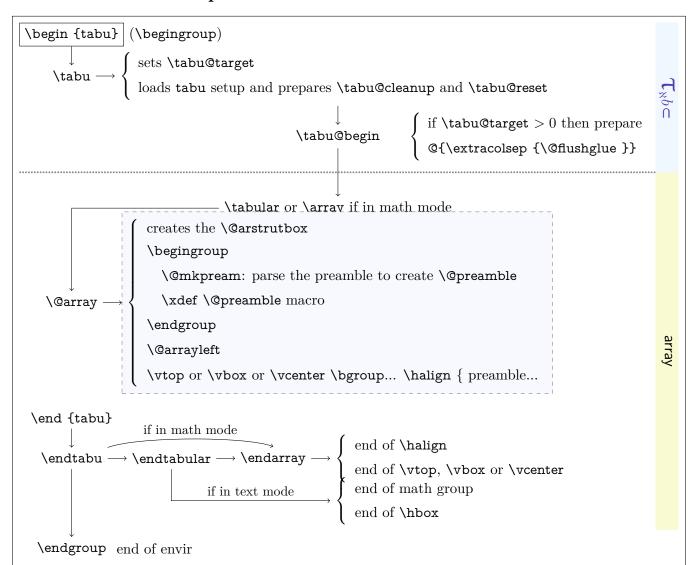
 $T_{\aleph}b \subset \text{requires array.sty}$ and varwidth.sty. The package namespace is tabu@.

```
1 (*package)
2 \NeedsTeXFormat{LaTeX2e}[2005/12/01]
3 \ProvidesPackage{tabu}[2011/02/26 v2.8 - flexible LaTeX tabulars (FC)]
4 \RequirePackage{array}[2008/09/09]
5 \RequirePackage{varwidth}[2009/03/30]
```

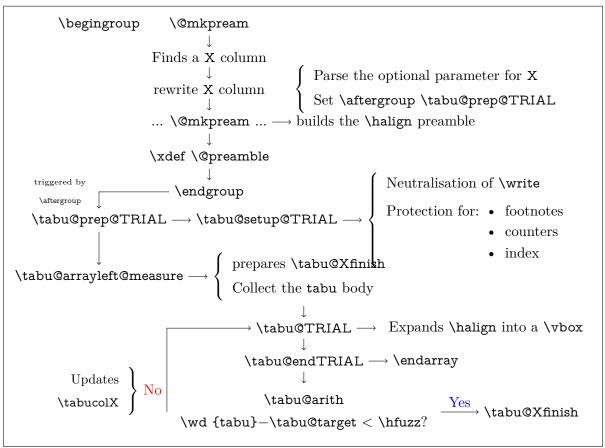
Minimal catcode acertaining for loading $\tau_{\aleph}b \subset$ in good conditions:

```
 6 \texttt{AtEndOfPackage} \{ \texttt{AtEnd } \texttt{AtEnd } \texttt{Qundefined} \} 
7 \let\tabu@AtEnd\@empty
8 \def\TMP@EnsureCode#1={%
      \edef\tabu@AtEnd{\tabu@AtEnd
10
                        \catcode#1 \the\catcode#1}%
11
      \catcode#1=%
12 }% \TMP@EnsureCode
13 \TMP@EnsureCode 33 = 12 % !
14 \TMP@EnsureCode 58 = 12 % : (for siunitx)
15 \TMP@EnsureCode124 = 12 % |
16 \TMP@EnsureCode 36 = 3 % $ = math shift
17 \TMP@EnsureCode 38 = 4 % & = tab alignmment character
18 \TMP@EnsureCode 32 = 10 % space
19 \TMP@EnsureCode 94 = 7 % ^
20 \TMP@EnsureCode 95 = 8 % _
```

11.5 Flow chart of expansion



tabu to with X column The important part of the job is made inside the dashed box above: \@mkpream expands the columns definitions, which can be user defined. Hopefully, it does its job inside a group, therefore a user-column can set a macro to be expanded \aftergroup. This implementation allows much modifications in the tabular preparation, without any change in the macros of array.sty.



tabu spread with X column

In the case of "tabu spread" with X columns, the process is the same as the one described for "tabu to" with X columns. However, the first trial is different because we have first to measure the *natural width* of the tabular. The process is the following:

- \tabu@target is first set to \linewidth (or \linegoal with the linegoal package option).
- The X column corresponds to a \vbox with \hsize fixed to \tabu@target.
- Inside this \vbox the cell content is written into a \hbox whose width is limited to \tabu@target. This \hbox is captured into the box register \tabu@box.
- At the end of the cell, the \badness of the \hbox is checked:
 - if the \badness is > 1000 then the text is too long and "tabu spread" is useless: tabu to \tabu@target give the same result.
 - Otherwise, we get the natural width of the cell content by: \setbox \tabu@box \hbox {\unhbox \tabu @box}
- At the end of the first trial, \tabu@spreadarith checks if:

- if not, then tabu to\tabu@target give the same result
- Otherwise, the target for tabu to will be:

width(tabular) + spread -
$$\sum_{i}$$
 natural widths $X_i + \underbrace{\text{Max}_i \left(\frac{\text{natural width } X_i}{\text{coef}_i} \right) \times \sum_{i} \text{coef}_i}_{i}$

minimal natural width that can be obtained

with the given coefs

11.6 Some constants

Here we define the constants used by $\tau_{\aleph}b \subset \tau_{\mathsf{FX}}$ registers and a few helper macros.

When working inside a tabular (ie.\halign) each cell is a T_FX group. Probably the most important property of each register defined here is whether it is global or not. A local register does not suffer, never, any global assignment.

T_EX registers

taburow

LATEX counter that globally stores the value of the current row. It is updated at \everyrow, rather than at \everycr⁸. \thetaburow expands to the (arabic) number.

This counter can be read by the user, but she must not change its value because it is used internally to store the height/depth of every row, for vertical spacing adjustment (and vertical leaders).

\tabu@nbcols

TEX counter that - locally - saves the total number of columns of the tabu. Special @ and! columns are not counted (they are not real columns for \h align, but only insertions into the preamble).

The value is used by \tabucline to ensure that the leader does not jut out over the last column...

\tabu@cnt

T_EX counter that – locally – stores the number of trials. Incidentally, it is also temporarily used to parse the width coefficient for X columns, during the rewriting process.

\tabu@Xcol

T_FX counter that - locally - stores the number of tabu X columns. Defined while rewriting the X token, it is used in the specification of the width of the column (\tabu@hsize {Rank of the X column}{coef}).

It is also used to store the natural width of X columns (in the cases of a negativ coefficient or if tabu spread is used).

\tabu@alloc \tabu@nested A global counter whose initial value (-1) is incremented for each nested tabular. The end of the outermost tabular globally resets the value to -1. \tabu@nested stores locally the value of \tabu@alloc and is therefore the "index" of the current tabular (the one that is actually in construction).

This influences the initialisation process (cf. \tabu@setup and \tabu@init).

\tabu@start \tabu@stop They are used locally by \tabucline and \everyrow while parsing the parameters: this is, for clarity, the local name for \@tempcnta and \@tempcntb.

```
21 \newcount \c@taburow
                               \def\thetaburow {\number\c@taburow}
22 \newcount \tabu@nbcols
23 \newcount \tabu@cnt
24 \newcount \tabu@Xcol
25 \let\tabu@start \@tempcnta
26 \let\tabu@stop \@tempcntb
27 \newcount \tabu@alloc \tabu@alloc=\m@ne
28 \newcount \tabu@nested
```

29 \def\tabu@alloc@{\global\advance\tabu@alloc \@ne \tabu@nested\tabu@alloc}

\tabu@target

TFX dimen that – locally – stores the tabu target (either "to" or "spread").

\tabu@spreadtarget

T_FX dimen that – locally – stores the tabu spread given by the user.

\tabu@naturalX

T_EX dimen that – globally – stores the total natural widths of the X columns, in the cases of negativ coefficients and/or tabu spread. The value is reset to 0pt at \everyrow, and maxima/minima are stored into the macro \tabu@naturalXmin and \tabu@naturalXmax: those are required for the algorithm of tabu spread (\tabu@spreadarith).

\tabucolX

T_FX dimen that – locally – stores the width corresponding to the preamble token X[1]: the standard width of X columns.

\tabu@DELTA

This is for clarity, the local name of \@tempdimc in \tabu@arith.

\tabu@thick \tabu@on \tabu@off

 $\mathsf{L}_{\aleph}b \subset [\text{rev.}2.8 \text{ release}] \otimes 2010 - 2011 \hookrightarrow \mathsf{FC}$

They are used locally by \tabu@getline, while parsing the parameters for a line specification. This is for clarity, the local name for \@tempdima, \@tempdimb and \@tempdimc.

^{8.} Package xcolor defines the \rownum TfX counter, which is globally updated at \everycr. Hence this \rownum counter is not reliable in case the user invokes \cline or \cmidrule for example...

```
\mathsf{T}_{\aleph}b \subset [\text{rev.}2.8 \text{ release}] \odot 2010 - 2011 \hookrightarrow \mathsf{FC}
```

```
30 \newdimen \tabu@target
31 \newdimen \tabu@spreadtarget
32 \newdimen \tabu@naturalX
33 \newdimen \tabucolX
34 \let\tabu@DELTA \@tempdimc
35 \let\tabu@thick \@tempdima
36 \let\tabu@on \@tempdimb
37 \let\tabu@off \@tempdimc
```

\tabu@Xsum

TEX dimen that – locally – stores the sum of all width coefficients for X columns. This is required to fix the initial value for \tabucolX and then in the algorithms (\tabu@arith and \tabu@arithnegcoef).

```
38 \newdimen \tabu@Xsum
```

\extrarowdepth

array.sty defines \extrarowheight as a T_EX dimen register: the extra height to be finally added to each row of a table. $T_Rb \subset$ defines \extrarowdepth in addition: the *extra depth*. Though \extrarowheight and \extrarowdepth can be set by the user, the official interface is \extrarowsep.

\abovetabulinesep \belowtabulinesep TEX dimensions \abovetabulinesep and \belowtabulinesep store the minimum allowed vertical space between the contents of the cells and their borders. Their values are ignored if non positive. Though they can be set by the user, the official interface is \tabulinesep.

The philosophy and the technics are similar to the one provided by the cellspace package. However, limitations of cellspace are lifted (nested tabu environments, use of colors... see the cellspace limitations in the revision history). $\tau_{\aleph}b \subset$ inserts only one strut per line, whose name is \@arstrut.

\tabustrutrule

The T_EX dimen \tabustrutrule is here only for debugging purpose: its value must be 0pt. It behaves mostly like T_EX primitive \overfullrule, and allow to see the struts introduced in the tabular, and to control vertical spacing. Setting \tabustrutrule to a positive value has no effect unless \tracingtabu is ≥ 3 . The official interface is \tracingtabu = 3.

```
39 \newdimen \extrarowdepth
40 \newdimen \abovetabulinesep
41 \newdimen \belowtabulinesep
42 \newdimen \tabustrutrule \tabustrutrule \z@
```

\tabu@thebody

This token stores – locally – the collected content of the tabu environment during the measuring process.

\tabu@footnotes

Token that globally stores the footnotes inside the tabu environment, for \insert does not work inside such a level of groupings...

```
43 \newtoks \tabu@thebody
44 \newtoks \tabu@footnotes
```

\tabu@box

Stores – loally – the whole tabu when an attempt to adjust X columns is performed.

\tabu@arstrutbox

While the \@arstrutbox may redefined globally at the end of each line (for vertical spacing adjustment), we define a new box and \let \@arstrutbox to be that box inside the tabu environment.

Hence, the \@arstrutbox used by other tabular environment does not suffer any modification.

\tabu@hleads
\tabu@vleads

Those boxes are used to built horizontal and vertical leaders. In order not to rebuilt the boxes every time a leader is inserted, the box is globally defined if a line style is specified (via | [line style] or \tabulinestyle {line style}.

```
45 \newsavebox \tabu@box
46 \newsavebox \tabu@arstrutbox
47 \newsavebox \tabu@hleads
48 \newsavebox \tabu@vleads
```

Switches

\iftabu@colortbl

The global switch \iftabu@colortbl is used by \rowfont when modifying the alignments, because colortbl changes the glues put inside the \halign preamble to make standard alignments. This switch is set At Begin Document.

\iftabu@siunitx

Global switch set \AtBeginDocument. true if siunitx package is detected.

\iftabu@measuring

This switch is somewhat magic in the sense that it has several meanings... It is temporarily set to true by \tabu@arith in the trial group, to say that the tabu did not reach its target yet. It is also set to true in the \@mkpream group when the first X column is encountered in the preamble. Finally, it is true in the trial **S** group when the outermost tabular is in strategy number 2 or number 3.

\iftabu@spread

A switch whether "tabu spread" is used or not. A nested tabu inside a X column whose coefficient is negative has a default target set to spread Opt.

\iftabu@negcoef

A switch set to true in case of negativ coef (natural width if less than X[coef]).

\iftabu@everyrow

A very important global switch: true when outside any tabu environment, true as well when inside a cell of a tabu, but globally set to false at \everycr and therefore inside any \noalign command. This allows to insert leaders (by \omit \span \omit \cr \noalign \{...\}) or first/last line corrections only once, even if \everycr is executed more than once.

\iftabu@long

Finally the swith \iftabu@long is set to true inside longtabu and to false inside tabu. This is convenient because some setup are slightly different between tabu and longtabu.

```
49 \newif \iftabu@colortbl
50 \newif \iftabu@siunitx
51 \newif \iftabu@measuring
52 \newif \iftabu@spread
53 \newif \iftabu@negcoef
54 \newif \iftabu@everyrow
55 \def\tabu@everyrowtrue {\global\let\iftabu@everyrow \iftrue}
56 \ensuremath{\mbox{\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$\mbox{$
57 \newif \iftabu@long
```

\tabu@rescan

\iftabuscantokens \iftabuscantokens is the switch for whether or not tabu will use \scantokens. Though the user can set \iffalse, the official interface is tabu*.

> It does not make sense to use \scantokens in a nested tabu: only the outermost tabu can use \scantokens, for the environment body must be collected with care!

\tabu@rescan is the helper macro for scanning tokens.

```
58 \newif \iftabuscantokens
59 \def\tabu@rescan {\tabu@verbatim \scantokens }
```

Some helper macros

\tabu@gobblespace

Two macros which are needed when scanning tokens with \futurelet.

\tabu@gobbletoken \tabu@gobbleX

This gobbles the character number 10 in ASCII (^^J in T_EX).

\tabu@ifenvir

Checks if the current environment is tabu or longtabu (for \multicolumn inside tabu).

\tabu@modulo

Computes the modulo (for \taburowcolors). The method is taken from H.O. intcalc package.

```
60 \def\tabu@gobblespace #1 {#1}
61 \def\tabu@gobbletoken #1#2{#1}
62 \def\tabu@gobbleX{\futurelet\@let@token \tabu@gobblex}
63 \def\tabu@gobblex{\if ^^J\noexpand\@let@token \expandafter\@gobble
64
                     \else\ifx \@sptoken\@let@token
65
                       \expandafter\tabu@gobblespace\expandafter\tabu@gobbleX
67 }% \tabu@gobblex
68 \def \tabu@X{^^J}
```

```
69 {\obeyspaces
70 \global\let\tabu@spxiii= % saves an active space (for \ifx)
71 \gdef\tabu@@spxiii{ }}
72 \def\tabu@ifenvir {% only for \multicolumn
73  \expandafter\tabu@if@nvir\csname\@currenvir\endcsname
74 }% \tabu@ifenvir
75 \def\tabu@ifenvir #1{\csname @\ifx\tabu#1first\else
76  \ifx\longtabu#1first\else
77  second\fi\fi oftwo\endcsname
78 }% \tabu@ifenvir
79 \def\tabu@modulo #1#2{\numexpr\ifnum\numexpr#1=\z@ 0\else #1-(#1-(#2-1)/2)/(#2)*(#2)\fi}
```

\tabu@strtrim Trimming spaces at low cost...

```
80 {\catcode'\&=3
81 \gdef\tabu@strtrim #1{% #1 = control sequence to trim
     \ifodd 1\ifx #1\@empty \else \ifx #1\space \else 0\fi \fi
            \let\tabu@c@l@r \@empty
83
                                      \let#1\@empty
84
            \expandafter \tabu@trimspaces #1&#1\@nnil
     \else
     \fi
85
86 }% \tabu@strtrim
87 \qdef\tabu@trimspaces #1&#2\@nnil{\let\tabu@c@l@r=#2\tabu@firstspace .#1& &#2}%
88 \gdef\tabu@firstspace #1#2#3 &{\tabu@lastspace #2#3&}
89 \def \abu@lastspace #1&#2&#3{\def #3{#1}%}
     \tabu@trimspaces #1&#3\@nnil}
91
92 }% \catcode
```

\tabu@sanitizearg

Sanitize an argument (babel compliant).

```
93 \def\tabu@sanitizearg #1#2{{%

94 \csname \ifcsname if@safe@actives\endcsname  % <babel>
95 @safe@activestrue\else

96 relax\fi \endcsname

97 \edef#2{#1}\tabu@strtrim#2\@onelevel@sanitize#2%

98 \expandafter}\expandafter\def\expandafter#2\expandafter{#2}%

99 }% \tabu@sanitizearg
```

\tabu@textbar

The character | may have a special category code inside the document, depending on the language setting or for example, | can be the delimiter shortcut for verbatim. We use \scantokens to allow an \ifx test even if the category code of | changes along the compilation.

```
100 \endinechar\end{ contokens {\def\:{|}}% } \\ 101 \endinechar\end{ contokens {\def\:{|}}% } \\ 102 \end{ contokens {\def\:{|}}% } \\ 102 \end{ contokens {\def\:{|}}% } \\ 103 \end{ contokens {\def\:{|}}% } \\ 104 \end{ contokens {\def\:{|}}% } \\ 105 \end{ contokens {\def\:{|}}% } \\ 106 \end{ contokens {\def\:{|}}% } \\ 107 \end{ contokens {\def\:{|}}% } \\ 108 \end{ contokens {\def\:{|}}% } \\ 109 \end{ contokens {\de
```

\tabu@everyrow@bgroup Commands like \everyrow, \taburulecolor, \tabulinestyle, \taburowcolors can be expanded either \tabu@everyrow@egrdipa cell or outside a tabu environment or at the end of a row, inside a \noalign group.

To avoid the insertion of an empty math atom (equivalent to \hootnote{hbox} to $0pt{}$) we open a semi-simple group rather than a math group if not in \note{hbox} is used to define the local-to-the-TEX-group setting (post-fixed by @L).

```
103 \def\tabu@everyrow@bgroup{\iftabu@everyrow \begingroup \else \noalign{\ifnum0='}\fi \fi}
104 \def\tabu@everyrow@egroup{%
105 \iftabu@everyrow \expandafter \endgroup \the\toks@
106 \else \ifnum0='{\fi}%
107 \fi
108 }% \tabu@everyrow@egroup
```

Rebuild the \@arstrutbox

```
\tabu@arstrut
\tabu@rearstrut
```

The macros rebuilds the $\c a \b)$. With the debug variants when $\t a \b)$ and $\t a \b)$.

```
109 \def\tabu@arstrut {\global\setbox\@arstrutbox \hbox{\vrule
       height \arraystretch \dimexpr\ht\strutbox+\extrarowheight
111
       depth \arraystretch \dimexpr\dp\strutbox+\extrarowdepth
       width \z@}%
113 }% \tabu@arstrut
114 \def\tabu@rearstrut {%
       \@tempdima \arraystretch\dimexpr\ht\strutbox+\extrarowheight \relax
       \@tempdimb \arraystretch\dimexpr\dp\strutbox+\extrarowdepth \relax
116
117
       \ifodd 1\ifdim \ht\@arstrutbox=\@tempdima
               \ifdim \dp\@arstrutbox=\@tempdimb 0 \fi\fi
119
           \tabu@mkarstrut
       \fi
120
121 }% \tabu@rearstrut
```

\tabu@DBG@arstrut

This is the "debug" version of \tabu@arstrut: used when \tracingtabu = 3 or more to show the struts inserted in the tabular.

```
122 \ensuremath{\mbox{def}\mbox{tabu@DBG $\#1{\ifdim}\tabustrutrule}\z@ \color{$\#1$\fi}}
123 \def\tabu@DBG@arstrut {\global\setbox\@arstrutbox
       \hbox to\z@{\hbox to\z@{\hss}
125
       {\tabu@DBG{cyan}\vrule
126
       height \arraystretch \dimexpr\ht\strutbox+\extrarowheight
127
       depth \z@
       width \tabustrutrule}\kern-\tabustrutrule
129
       {\tabu@DBG{pink}\vrule
130
       height \z@
       depth \arraystretch \dimexpr\dp\strutbox+\extrarowdepth
131
132
       width \tabustrutrule}}}%
133 }% \tabu@DBG@arstrut
```

\tabu@save@decl

No inversion on tokens in the tabu preamble, when not in math mode.

\tabu@finalstrut

 $\mathsf{L}_{\aleph}b \subset [\text{rev.2.8 release}] \odot$

139 \def\tabu@finalstrut #1{\unskip\ifhmode\nobreak\fi\vrule height\z@ depth\z@ width\z@}

Disable some commands during trials

\tabuDisableCommands Following the model of hyperref \pdfstringdefDisableCommands, \tabuDisableCommands allow the user to change the definition of some commands during the trial loops, by the mean of a hook to be expanded by \tabu@setstrategy.

```
140 \end{tabu} E commands {\g@addto@macro\tabu@trialh@@k } 141 \eltabu@trialh@@k \end{tabu}
```

\[\tabu@nowrite \text{A trick (from the TFX-book) to forbidd \write when a trial is done on the \halign.

\tabu@noxfootnotes Disable footnotes during trials.

```
142 \def\tabu@nowrite #1#{{\afterassignment}\toks@}
143 \let\tabu@write\write
144 \let\tabu@immediate\immediate
145 \def\tabu@WRITE{\begingroup
146 \def\immediate\write{\aftergroup\endgroup}
```

```
147
                      \tabu@immediate\tabu@write}%
148 }% \tabu@WRITE
149 \expandafter\def\expandafter\tabu@GenericError\expandafter{%
                          \expandafter\tabu@WRITE\GenericError}
151 \def\tabu@warn{\tabu@WRITE\PackageWarning{tabu}}
152 \def\tabu@noxfootnote [#1]{\@gobble}
```

\tabu@nocolor \tabu@norowcolor

For optimisation purpose, color changes are deactivated during trials, for they do not affect the measures.

```
153 \def\tabu@nocolor #1#{\@gobble}
154 \newcommand*\tabu@norowcolor[2][]{}
```

siunitx S and s columns management

\tabu@maybesiunitx A macro that encloses the definition of \tabu@celllalign, in order to check if the column is a siunitx S (or s) column, and neutralise the setup of \rowfont in this case, for signitx provides its own key=value options to set fonts inside S (or s) columns.

```
155 \def \tabu@maybesiunitx #1{\def \tabu@temp{#1}%}
                              \futurelet\@let@token \tabu@m@ybesiunitx}
157 \def\tabu@m@ybesiunitx \#1{\def\tabu@m@ybesiunitx } {%
158
       \ifx #1\@let@token \let\tabu@cellleft \@empty \let\tabu@cellright \@empty \fi
       \tabu@temp}% \tabu@m@ybesiunitx
159
160 }\expandafter\tabu@m@ybesiunitx \csname siunitx_table_collect_begin:Nn\endcsname
161 \def\tabu@celllalign@def #1{\def\tabu@celllalign{\tabu@maybesiunitx{#1}}}%
```

11.7 Rules, colors and vertical adjustment

\extrarowsep and \tabulinesep

\extrarowsep

\extrarowsep makes the assignment for both \extrarowheight and \extrarowdepth.

The macro may be prefixed by \global.

```
162 \ensuremath{\mbox{\mbox{$1$}}} extrarowsep{\ensuremath{\mbox{\mbox{$4$}}} edef\ensuremath{\mbox{$4$}} extra{\ensuremath{\mbox{$4$}}} estrate {\ensuremath{\mbox{$4$}}} est
163
                 \iftabu@everyrow
                                                                  \aftergroup\tabu@Gextra
164
                 \else
                                                                   \aftergroup\tabu@n@Gextra
                 \fi
165
                 \@ifnextchar={\tabu@gobbletoken\tabu@extra} \tabu@extra
167 }% \extrarowsep
168 \def\tabu@extra {\@ifnextchar_%
                 {\tabu@gobbletoken{\tabu@setextra\extrarowheight \extrarowdepth}}
169
170
                 {\ifx ^\@let@token \def\tabu@temp{%
171
                                     \tabu@gobbletoken{\tabu@setextra\extrarowdepth \extrarowheight}}%
                                     \let\tabu@temp \@empty
172
173
                                     \afterassignment \tabu@setextrasep \extrarowdepth
174
                 \fi \tabu@temp}%
175 }% \tabu@extra
177 \def\tabu@extr@ #1#2{\@ifnextchar^%
                  {\tabu@gobbletoken{\tabu@setextra\extrarowdepth \extrarowheight}}
178
179
                 {\ifx _\@let@token \def\tabu@temp{%
                                     \tabu@gobbletoken{\tabu@setextra\extrarowheight \extrarowdepth}}%
180
181
                                     \let\tabu@temp \@empty
                 \else
182
                                     \tabu@Gsave \tabu@G@extra \tabu@C@extra \extrarowheight \extrarowdepth
                 \fi \tabu@temp}%
184 }% \tabu@extr@
185 \def\tabu@setextrasep {\extrarowheight=\extrarowdepth
                 \tabu@Gsave \tabu@G@extra \tabu@C@extra \extrarowheight \extrarowdepth
186
187 }% \tabu@setextrasep
188 \def\tabu@Gextra{\ifx \tabu@Gextra\@empty \else {\tabu@Rextra}\fi}
189 \def\tabu@n@Gextra{\ifx \tabu@G@extra\@empty \else \noalign{\tabu@Rextra}\fi}
```

```
190 \def\tabu@Rextra{\tabu@Grestore \tabu@G@extra \tabu@C@extra}
191 \let\tabu@C@extra \z@
192 \let\tabu@G@extra \@empty
```

\tabulinesep

\tabulinesep makes the assignment for both \abovetabulinesep and \belowtabulinesep.

The macro may be prefixed by \global.

```
193 \newcommand*\tabulinesep{\edef\tabu@C@linesep{\the\numexpr\tabu@C@linesep+1}%
194
       \iftabu@everyrow
                            \aftergroup\tabu@Glinesep
195
       \else
                            \aftergroup\tabu@n@Glinesep
196
       \fi
       \@ifnextchar={\tabu@gobbletoken\tabu@linesep} \tabu@linesep
197
198 }% \tabulinesep
199 \def\tabu@linesep {\@ifnextchar_%
200
       {\tabu@gobbletoken{\tabu@setsep\abovetabulinesep \belowtabulinesep}}
201
       {\ifx ^\@let@token \def\tabu@temp{%
               \tabu@gobbletoken{\tabu@setsep\belowtabulinesep \abovetabulinesep}}%
202
        \else \let\tabu@temp \@empty
203
204
               \afterassignment \tabu@setlinesep \abovetabulinesep
205
        \fi \tabu@temp}%
206 }% \tabu@linesep
207 \def \tabu@setsep \#1\#2 {\def \tabu@temp{\tabu@setsep\#1\#2} \after assignment \tabu@temp\#2} }
208 \def\tabu@sets@p #1#2{\ensuremath{\mbox{\circle}}}
       {\tabu@gobbletoken{\tabu@setsep\belowtabulinesep \abovetabulinesep}}
209
       {\ifx _\@let@token \def\tabu@temp{%
210
               \tabu@gobbletoken{\tabu@setsep\abovetabulinesep \belowtabulinesep}}%
211
212
               \let\tabu@temp \@empty
       \else
213
                \tabu@Gsave \tabu@G@linesep \tabu@C@linesep \abovetabulinesep \belowtabulinesep
       \fi \tabu@temp}%
215 }% \tabu@sets@p
216 \def\tabu@setlinesep {\belowtabulinesep=\abovetabulinesep
       \tabu@Gsave \tabu@G@linesep \tabu@C@linesep \abovetabulinesep \belowtabulinesep
218 }% \tabu@setlinesep
219 \def\tabu@Glinesep{\ifx \tabu@G@linesep\@empty \else {\tabu@Rlinesep}\fi}
220 \def\tabu@n@Glinesep{\ifx \tabu@G@linesep\@empty \else \noalign{\tabu@Rlinesep}\fi}
221 \def\tabu@Rlinesep{\tabu@Grestore \tabu@G@linesep} \tabu@C@linesep}
222 \let\tabu@C@linesep \z@
223 \let\tabu@G@linesep \@empty
```

\tabu@Gsave Utility macros to implement the possibility to prefix a macro by \global.

\tabu@Grestore

```
224 \def\tabu@Gsave #1#2#3#4{\xdef#1{#1%
     \label{local_state} $$ \toks#2{\toks\the\currentgrouplevel{\global#3\\the#3\global#4\\the#4}}}
226 }% \tabu@Gsave
227 \def\tabu@Grestore#1#2{%
     \ifcat$\the\toks\currentgrouplevel$\else
229
230
        \the\toks\currentgrouplevel
231
     \fi
233 }% \tabu@Grestore
```

Setting code for every row

\everyrow

As long as tabu needs to execute some code at \everycr, it's not difficult to provide a command to give the user the opportunity to execute its own arbitrary code. However, \everyrow will be used almost only with \hline (or \tabucline or \midrule).

\everyrow can be changed anywhere inside the tabu: at the end of a row, or even inside a cell.

The rows LATEX counter taburow must not be changed by the user!.

The settings are saved in a "locally-global" way...

```
234 \newcommand*\everyrow{\tabu@everyrow@bgroup
                         \tabu@start \z@ \tabu@stop \z@ \tabu@evrstartstop
235
236 }% \everyrow
237 \def\tabu@evrstartstop {\@ifnextchar^%
       {\afterassignment \tabu@evrstartstop \tabu@stop=}%
239
       {\ifx ^\@let@token
               \afterassignment\tabu@evrstartstop \tabu@start=%
240
        \toks@
241
242
        \fi}%
243 }% \tabu@evrstartstop
244 \def\tabu@everyr@w {%
245
       \xdef\tabu@everyrow{%
           \noexpand\tabu@everyrowfalse
246
           \let\noalign \relax
247
248
           \noexpand\tabu@rowfontreset
249
           \iftabu@colortbl \noexpand\tabu@rc@ \fi % \taburowcolors
           \let\noexpand\tabu@docline \noexpand\tabu@docline@evr
250
251
           \the\toks@
252
           \noexpand\tabu@evrh@@k
253
           \noexpand\tabu@rearstrut
254
           \qlobal\advance\c@taburow \@ne}%
255
       \iftabu@everyrow \toks@\expandafter
           {\expandafter\def\expandafter\tabu@evr@L\expandafter{\the\toks@}\ignorespaces}%
256
257
       \else \xdef\tabu@evr@G{\the\toks@}%
258
       \fi
       \tabu@everyrow@egroup
259
260 }% \tabu@everyr@w
261 \def\tabu@evr {\def\tabu@evrh@@k}
                                             % for internal use only
262 \tabu@evr{}
```

Setting line styles and colors

\newtabulinestyle

\newtabulinestyle {style=spec.,style=spec,style=spec}

All the job is done by \tabu@getline. New line style specification are always defined globally, and can be overwritten without warning...

```
263 \newcommand*\newtabulinestyle [1] {%
       {\@for \@tempa :=#1\do{\expandafter\tabu@newlinestyle \@tempa==\@nil}}%
265 }% \newtabulinestyle
266 \def\tabu@newlinestyle #1=#2=#3\@nil{\tabu@getline {#2}%
267
       \tabu@sanitizearg {#1}\@tempa
268
       \ifodd 1\ifx \@tempa\@empty \ifdefined\tabu@linestyle@ 0 \fi\fi
269
       \global\expandafter\let
           \csname tabu@linestyle@\@tempa \endcsname =\tabu@thestyle \fi
271 }% \tabu@newlinestyle
```

\tabulinestyle \tabul

The job is done by \tabu@getline. The settings as usual, are stored in a "locally-global" way...

272 \newcommand*\tabulinestyle [1]{\tabu@everyrow@bgroup \tabu@getline{#1}%

```
273
       \iftabu@everyrow
274
            \toks@\expandafter{\expandafter \def \expandafter
275
                        \tabu@ls@L\expandafter{\tabu@thestyle}\ignorespaces}%
276
           \gdef\tabu@ls@{\tabu@ls@L}%
277
       \else
278
           \global\let\tabu@ls@G \tabu@thestyle
279
            \gdef\tabu@ls@{\tabu@ls@G}%
280
       \fi
281
       \tabu@everyrow@egroup
282 }% \tabulinestyle
```

\taburulecolor

colortbl provides \arrayrulecolor, but the definition is global and must be restores manually after the table. \taburulecolor works with the same scheme as \everyrow: even if the definition of the rules colors must be global (because we it can be changed inside the tabular) the value is not restored globally at the end of the environment.

Instead, \tabu@arc@L stores locally the color definition (ie.its definition is relative to the group level before the entry inside the tabu environment).

This is the same for \doublerulesepcolor (which may be given as an optional argument to \taburulecolor): colortbl makes the definition global, while $\mathcal{T}_{\aleph}b \subset$ keeps grouping level into mind ("locally-global" settings).

```
283 \newcommand*\taburulecolor{\tabu@everyrow@bgroup \tabu@textbar \tabu@rulecolor}
284 \ensuremath{\mbox{def}\mbox{tabu@rulecolor}} #1{\toks@{}}%
                   \def\tabu@temp #1##1#1{\tabu@ruledrsc{##1}}\@ifnextchar #1%
286
                                                                                                                                                  \tabu@temp
287
                                                                                                                                          \tabu@rulearc
288 }% \tabu@rulecolor
289 \def\tabu@ruledrsc #1{\edef\tabu@temp{#1}\tabu@strtrim\tabu@temp
                  \ifx \tabu@temp\@empty \def\tabu@temp{\tabu@rule@drsc@ {}{}}%
                  \else \edef\tabu@temp{\noexpand\tabu@rule@drsc@ {}{\tabu@temp}}%
291
292
                  \fi
293
                  \tabu@temp
294 }% \tabu@ruledrsc@
295 \def\tabu@ruledrsc@
                                                                  #1#{\tabu@rule@drsc@ {#1}}
296 \def \tabu@rule@drsc@ #1#2{%}
297
                   \iftabu@evervrow
298
                             \toks@{\def\CT@drsc@{\color #1{#2}}}%
299
                             \else
300
                             \fi
301
                  \else
                             \  \ \\#1#2\\\global\let\CT@drsc@ \relax
303
                                                               \gdef\CT@drsc@{\color #1{#2}}%
304
                             \fi
                  \fi
305
                  \tabu@rulearc
306
307 }% \tabu@rule@drsc@
308 \def\tabu@rulearc
                                                               #1#{\tabu@rule@arc@ {#1}}
309 \def \tabu@rule@arc@ #1#2{%}
310
                  \iftabu@everyrow
                             \fint \fin
311
                             \else
                                                               312
313
                             \fi
                             \toks@\expandafter{\the\toks@
315
                                       \let\tabu@arc@L \CT@arc@
                                       \let\tabu@drsc@L \CT@drsc@
316
                                       \ignorespaces}%
317
318
                  \else
319
                             \ifx \\#1#2\\\qdef\CT@arc@{}%
                                                               \gdef\CT@arc@{\color #1{#2}}%
320
                             \else
```

```
321 \fi
322 \global\let\tabu@arc@G \CT@arc@
323 \global\let\tabu@drsc@G \CT@drsc@
324 \fi
325 \tabu@everyrow@egroup
326 }% \tabu@rule@arc@
```

\taburowcolors

 $\taburowcolors {number} \langle number \rangle \{first color .. last color \}$

The aim of the game is to define the process that will be executed at \everyrow.

After that, the usual process for "locally-global" settings is plugged into \tabu@cleanup and \tabu@reset...

```
327 \def\taburowcolors {\tabu@everyrow@bgroup \@testopt \tabu@rowcolors 1}
328 \ensuremath{\mbox{\mbox{$1$}}} \#2 \ensuremath{\mbox{\mbox{\mbox{$4$}}}} \#2 \ensuremath{\mbox{\mbox{\mbox{$4$}}}} \#2 \ensuremath{\mbox{\mbox{\mbox{$4$}}}} \#2 \ensuremath{\mbox{\mbox{$4$}}} \#2 \ensuremath{\mbox{\mbox{$4$}}} \#2 \ensuremath{\mbox{$4$}}} \#2 \ensuremath{\mbox{\mbox{$4$}}} \#2 \ensuremath{\mbox{$4$}} \#2 \ensuremath{\mbox{$4$}}} \#2 \ensuremath{\mbox{$4$}} \#2 \ensuremath{\mbox{$4$}}} \#2 \ensuremath{\mbox{$4$}} \#2 \ensuremath{\mbox{$4$}}} \#2 \ensuremath{\mbox{$4$}} \#2 \ensuremath{\mbox{$4$}}} \#2 \ensuremath{\mbox{$4$}} \#2 \ensuremath{\mbox{$4$}} \#2 \ensuremath{\mbox{$4$}}} \#2 \ensuremath{\mbox{$4$}} \#2 \ensuremath{\mbox{$4$}}} \#2 \ensuremath{\mbox{$4$}} \#2 \ensuremath{\mbox{$
329 \def\tabu@rowc@lors #1#2#3{%
330
                    \toks@{}\@defaultunits \count@
                                                                                                                              =\number0#2\relax \@nnil
331
                                             \@defaultunits \tabu@start =\number0#1\relax \@nnil
332
                    \ifnum \count@<\tw@ \count@=\tw@ \fi
333
                    \advance\tabu@start \m@ne
334
                    \ifnum \tabu@start<\z@ \tabu@start \z@ \fi
335
                    \tabu@rowcolorseries #3\in@..\in@ \@nnil
336 }% \tabu@rowcolors
337 \def\tabu@rowcolorseries #1..#2\in@ #3\@nnil {%
                     \ifx \in@#1\relax
339
                                \iftabu@everyrow \toks@{\def\tabu@rc@{}\let\tabu@rc@L \tabu@rc@}%
                                                        \gdef\tabu@rc@{}\global\let\tabu@rc@G \tabu@rc@
340
                                \else
                                \fi
341
342
                    \else
343
                                \ifx \\#2\\\tabu@rowcolorserieserror \fi
                                \tabu@sanitizearg{#1}\tabu@temp
344
345
                                \tabu@sanitizearg{#2}\@tempa
346
                                \advance\count@ \m@ne
347
                    \iftabu@everyrow
                                \def\tabu@rc@ ##1##2##3##4{\def\tabu@rc@{%
348
349
                                            \ifnum ##2=\c@taburow
                                                        \definecolorseries{tabu@rcseries@\the\tabu@nested}{rgb}{last}{##3}{##4}\fi
350
351
                                            \ifnum \c@taburow<##2 \else
                                                        \infty \time \ti
352
                                                                    \resetcolorseries[{##1}]{tabu@rcseries@\the\tabu@nested}\fi
353
354
                                                        \xqlobal\colorlet{tabu@rc@\the\tabu@nested}{tabu@rcseries@\the\tabu@nested!!+}%
                                                        \rowcolor{tabu@rc@\the\tabu@nested}\fi}%
356
                                }\edef\x{\noexpand\tabu@rc@
                                                                                                                                                 {\the\count@}
357
                                                                                                                                     {\the\tabu@start}
                                                                                                                                                    {\tabu@temp}
358
359
                                                                                                                                                             {\@tempa}%
360
                                \toks@\expandafter{\expandafter\def\expandafter\tabu@rc@\expandafter{\tabu@rc@}}%
361
362
                                \toks@\expandafter{\the\toks@ \let\tabu@rc@L \tabu@rc@ \ignorespaces}%
363
                                            % inside \noalign
364
                                \definecolorseries{tabu@rcseries@\the\tabu@nested}{rgb}{last}{\tabu@temp}{\@tempa}%
                                \expandafter\resetcolorseries\expandafter[\the\count@]{tabu@rcseries@\the\tabu@nested}%
365
366
                                \xqlobal\colorlet{tabu@rc@\the\tabu@nested}{tabu@rcseries@\the\tabu@nested!!+}%
                                \let\noalign \relax \rowcolor{tabu@rc@\the\tabu@nested}%
                                \def\tabu@rc@ ##1##2{\gdef\tabu@rc@{%
368
                                            \in \tabu@modulo {\c@taburow-##2}{##1+1}=\end{modulo}
369
                                                        \resetcolorseries[{##1}]{tabu@rcseries@\the\tabu@nested}\fi
370
                                            \xqlobal\colorlet{tabu@rc@\the\tabu@nested}{tabu@rcseries@\the\tabu@nested!!+}%
371
372
                                             \rowcolor{tabu@rc@\the\tabu@nested}}%
                                 }\edef\x{\noexpand\tabu@rc@{\the\count@}{\the\c@taburow}}\x
373
```

```
374 \global\let\tabu@rc@G \tabu@rc@
375 \fi
376 \fi
377 \tabu@everyrow@egroup
378 }% \tabu@rowcolorseries
379 \tabuDisableCommands {\let\tabu@rc@ \@empty }
380 \def\tabu@rowcolorserieserror {\PackageError{tabu}}
381 {Invalid syntax for \string\taburowcolors
382 \MessageBreak Please look at the documentation!}\@ehd
383 }% \tabu@rowcolorserieserror
```

\tabureset

Simply – and locally – reset the default values for \tabulinesep (0pt), \extrarowsep (0pt), \extrarowsep (0pt), \tabulinestyle \{\}, \everyow \{\} and \taburulecolor []\{\}.

```
384 \newcommand*\tabureset {%
385 \tabulinesep=\z@ \extrarowsep=\z@ \extratabsurround=\z@
386 \tabulinestyle{}\everyrow{}\taburulecolor||{}\taburowcolors{}%
387 }% \tabureset
```

Parsing line styles

\tabu@getline

This macro parses a line specification argument of the form:

3pt BlanchedAlmond on 4pt Crimsom off 2pt ForestGreen

Note that Crimson will overwrite BlanchedAlmond in this case: the color for the line dash may be specified after the line width or after the line dash length.

The process uses \scantokens on the argument given by the user, which is first expanded in a context where the babel switch \if@save@actives is set to true. Then \scantokens is used on the argument in a group where the letter "o" is active, and defined to be a macro which rewrites the line specification. Incidentally, the comma is active too, and expands to a space. This way the initial argument is "genetically modified", so that it becomes very easy to assign dimensions (thickness, dash length and gap length) and colors separately.

For example: 3pt BlanchedAlmond on 4pt Crimson will be expanded in a context where "o" is active (and equal to \tabu@oxiii, the xiii suffix means "active" *ie*.\catcode = 13).

Then the "o" in Blanched Almond is rewritten as follow:

- 1. "o" sees "n" after itself, then it expands \tabu@onxiii.
- \tabu@onxiii sees a character whose catcode is not other, then the rewriting process is aborted, and "ond" is rewritten as "ond" where the "o" is not active but the usual letter "o".

The next "o" is rewritten as follow:

- 1. "o" sees "n" after itself, then it expands \tabu@onxiii.
- 2. \tabu@onxiii sees a space (which is active): it calls back itself again,
- 3. \tabu@onxiii sees a character whose catcode is other: then the sequence "on_□3" is rewritten as:

```
"\tabu@ \tabu@on =4pt Crimson"
```

Finally the whole argument is rewritten as:

\tabu@

- 1. Assign the corresponding dimension (thickness, dash length or gap length).
- 2. Collect the rest until the next \tabu@, trim spaces and check if the color exists.

Limitation: A color name must not contain a sequence that matches on of the patterns:

```
...on\langle a \ character \ of \ category \ 12 \rangle... or ...off\langle a \ character \ of \ category \ 12 \rangle...
```

But this "limitation" is not too heavy, I suppose...

438

439

The result is \tabu@thestyle: a tabu line style to be used to rewrite a | column, for \tabucline.

We use locally the LATEX defined dimen registers \@tempdima, \@tempdimb and \@tempdimc. For clarity, their names are \tabu@thick, \tabu@on and \tabu@off here...

```
388 \def\tabu@getline #1{\begingroup
389
       \csname \ifcsname if@safe@actives\endcsname
                                                              % <babel>
390
                          @safe@activestrue\else
391
                          relax\fi
                                          \endcsname
392
       \edef\tabu@temp{#1}\tabu@sanitizearg{#1}\@tempa
       \let\tabu@thestyle \relax
393
394
       \ifcsname tabu@linestyle@\@tempa \endcsname
               \edef\tabu@thestyle{\endgroup
395
396
                    \def\tabu@thestyle{\expandafter\noexpand
397
                        \csname tabu@linestyle@\@tempa\endcsname}%
398
                }\tabu@thestyle
               \expandafter\tabu@definestyle \tabu@temp \@nil
399
       \else
       \fi
400
401 }% \tabu@getline
```

\tabu@definestyle

```
Here is the \scantokens stuff.
403
       \tabu@thick \maxdimen \tabu@on \maxdimen
                                                    \tabu@off \maxdimen
404
       \let\tabu@c@lon \@undefined \let\tabu@c@loff \@undefined
405
       \ifodd 1\ifcat .#1\else\ifcat\relax #1\else 0\fi\fi % catcode 12 or non expandable cs
               \def\tabu@temp{\tabu@getparam{thick}}%
406
               \def\tabu@temp{\tabu@getparam{thick}\maxdimen}%
407
       \else
408
       \fi
409
       18
           \let\tabu@ \relax
410
411
           \def\:{\obeyspaces \tabu@oXIII \tabu@commaXIII \edef\:}% (space active \: happy ;-))
412
           \scantokens{\:{\tabu@temp #1#2 \tabu@\tabu@}}%
413
                            \expandafter}\expandafter
414
                                    \def\expandafter\:\expandafter{\:}% line spec rewritten now ;-)
415
       \def\;{\def\:}%
       \scantokens\expandafter{\expandafter\;\expandafter{\:}}% space is now inactive (catcode 10)
416
417
       \let\tabu@ \tabu@getcolor
                                     \: %
                                            all arguments are ready now ;-)
       \ifdefined\tabu@c@lon \else \let\tabu@c@lon\@empty \fi
418
419
       \ifx \tabu@c@lon\@empty \def\tabu@c@lon{\CT@arc@}\fi
420
       \ifdefined\tabu@c@loff \else \let\tabu@c@loff \@empty
                                                                      \fi
       \ifdim \tabu@on=\maxdimen \ifdim \tabu@off<\maxdimen
422
                                      \tabu@on \tabulineon
                                                                   \fi\fi
423
       \ifdim \tabu@off=\maxdimen \ifdim \tabu@on<\maxdimen
                                      \tabu@off \tabulineoff
424
                                                                   \fi\fi
       \ifodd 1\ifdim \tabu@off=\maxdimen \ifdim \tabu@on=\maxdimen 0 \fi\fi
425
426
                \in@true
                            % <leaders>
                            % <rule>
427
       \else
               \in@false
428
429
       \ifdim\tabu@thick=\maxdimen \def\tabu@thick{\arrayrulewidth}%
       \else
                                    \edef\tabu@thick{\the\tabu@thick}%
430
       \fi
431
432
       \edef \tabu@thestyle ##1##2{\endgroup
            \def\tabu@thestyle{%
433
434
               \ifin@ \noexpand\tabu@leadersstyle {\tabu@thick}
435
                                                     {\theta \in \text{tabu@on} \{ \#1 \}}
                                                     {\theta \in \mathbb{R} \ {\#2}}
436
437
                \else
                        \noexpand\tabu@rulesstyle
```

{##1\vrule width \tabu@thick}%

{##1\leaders \hrule height \tabu@thick \hfil}%

```
\mathsf{T}_{\aleph}b \subset [\text{rev.2.8 release}] \otimes 2010 - 2011 \Leftrightarrow \mathsf{F}
```

```
440 \fi}%

441 }\expandafter \expandafter

442 \expandafter \tabu@thestyle \expandafter

443 \expandafter \expandafter

444 \{\expandafter\tabu@c@lon\expandafter}\expandafter{\tabu@c@loff}%

445}% \tabu@definestyle
```

\tabu@onxiii
\tabu@ofxiii
\tabu@offiii

We have to define the active "o" character, which looks for the next tokens, trying to find a pattern like $\operatorname{on}\langle\operatorname{category}\ 12\rangle$ or $\operatorname{off}\langle\operatorname{category}\ 12\rangle$ (possibly with – active – spaces between on or off and the next character of catcode 12).

```
446 {\catcode'\0=\active \lccode'\0='\o \catcode'\,=\active
447
       \lowercase{\gdef\tabu@oXIII {\catcode \o=\active \let O=\tabu@oxiii}}
       \gdef\tabu@commaXIII {\catcode \\, =\active \let , =\space}
448
449 }% \catcode
450 \def \tabu@oxiii #1{%}
451
       \ifcase \ifx n#1\z@ \else
                 \fi\fi
453
                 \t.w@
             \expandafter\tabu@onxiii
454
455
       \or
             \expandafter\tabu@ofxiii
       \else o%
456
457
       \fi#1}%
458 \def\tabu@onxiii #1#2{%
       \ifcase \ifx !#2\tw0
459
                                         \else
460
                 \left(\frac{2}{z}\right) \sim \left(\frac{z}{z}\right)
                 \ifx \tabu@spxiii#2\@ne\else
461
                                     \fi\fi\fi
462
                 \t.w@
463
             \tabu@getparam{on}#2\expandafter\@gobble
             \expandafter\tabu@onxiii
                                           % (space is active)
465
       \else o\expandafter\@firstofone
       \fi{#1#2}}%
466
467 \def\tabu@ofxiii #1#2{%
       \ifx #2f\expandafter\tabu@offxiii
468
469
       \else
               o\expandafter\@firstofone
       \fi{#1#2}}
470
471 \def\tabu@offxiii #1#2{%
472
       \ifcase \ifx !#2\tw@
                                         \else
                \ifcat.\noexpand#2\z@
473
                                         \else
                \ifx\tabu@spxiii#2\@ne \else
474
475
                                     \fi\fi\fi
476
             \tabu@getparam{off}#2\expandafter\@gobble
             \expandafter\tabu@offxiii
477
                                          % (space is active)
478
       \else o\expandafter\@firstofone
479
       \fi{#1#2}}
```

\tabu@getparam

The rewritten stuff.

```
480 \end{figure} 1 {\columnwidth} 1 {\
```

_tabu@getcolor

\tabu@ \tabu@on = $\langle \beta pt \rangle$ Crimson\tabu@

\tabu@getcolor first makes the assignment to \tabu@on and then looks for the color name which might have been placed before the next \tabu@.

```
481 \def\tabu@getcolor #1{% \tabu@ <- \tabu@getcolor after \edef

482 \ifx \tabu@#1\else % no more spec

483 \let\tabu@theparam=#1\afterassignment \tabu@getc@l@r #1\fi

484 }% \tabu@getcolor

485 \def\tabu@getc@l@r #1\tabu@ {%

486 \def\tabu@temp{#1}\tabu@strtrim \tabu@temp
```

```
487
       \ifx \tabu@temp\@empty
       \else%\ifcsname \string\color@\tabu@temp \endcsname
                                                              % if the color exists
489
           \ifx \tabu@theparam \tabu@off
                                            \let\tabu@c@loff \tabu@c@l@r
                                             \let\tabu@c@lon \tabu@c@l@r
           \else
490
           \fi
491
492
       %\else \tabu@warncolour{\tabu@temp}%
493
       \fi%\fi
       \tabu@ % next spec
494
495 }% \tabu@getc@l@r
496 \def\tabu@warncolour #1{\PackageWarning{tabu}
       {Color #1 is not defined. Default color used}%
498 }% \tabu@warncolour
```

\tabu@rulesstyle

\tabu@leadersstyle When a style is executed, it expands either \tabu@leadersstyle or \tabu@rulesstyle depending on whether or not it contains leaders (dashed lines) or simple rules (solid lines): TFX internals allow to insert solid lines easily inside a tabular, while inserting leaders is more complex.

> \tabu@leadersstyle eventually rebuilds the (horizontal and vertical) leaders boxes, and then define two macros: \tabu@thevleaders and \tabu@thehleades, suitable to draw vertical and horizontal lines respectively. Incidentally, \tabu@leaders is defined to be the parameters for the leaders.

> \tabu@rulesstyle only defines the two macros \tabu@thevrule and \tabu@thehrule. The control sequence \tabu@leaders is undefined so that we know if the style contains a leader or a rule.

```
499 \def\tabu@leadersstyle #1#2#3#4#5{\def\tabu@leaders{{#1}{#2}{#3}{#4}{#5}}%
500
       \ifx \tabu@leaders\tabu@leaders@G \else
501
                   \tabu@LEADERS{#1}{#2}{#3}{#4}{#5}\fi
502 }% \tabu@leadersstyle
503 \def\tabu@rulesstyle #1#2{\let\tabu@leaders \@undefined
           \gdef\tabu@thevrule{#1}\gdef\tabu@thehrule{#2}%
505 }% \tabu@rulesstyle
```

\tabu@LEADERS

Here the two leaders boxes \tabu@hleads and \tabu@vleads are built, as well as the leaders macros \tabu@thehleaders and \tabu@thevleaders.

```
507
     {\let\color \tabu@color % => during trials -> \color = \tabu@nocolor
508
                             but the leaders boxes should have colors !
     \def\@therule{\vrule}\def\@thick{height}\def\@length{width}%
509
510
     \def\@box{\hbox}\def\@unbox{\unhbox}\def\@elt{\wd}%
     \label{leads} $$ \left( \skip_{\skip} \right) \ef\tabuleleads_{\tabuleleads} $$
511
512
     \tabu@l@@d@rs {#1}{#2}{#3}{#4}{#5}%
     \global\let\tabu@thehleaders \tabu@theleaders
513
514
     1 %
515
     516
517
     \def\@skip{\vskip}\def\@ss{\vss}\def\tabu@leads{\tabu@vleads}%
518
519
     \tabu@l@@d@rs {#1}{#2}{#3}{#4}{#5}%
     \global\let\tabu@thevleaders \tabu@theleaders
520
521
522
     \gdef\tabu@leaders@G{{#1}{#2}{#3}{#4}{#5}}%
523
     } 응
524 }% \tabu@LEADERS
525 \det tabu@therule #1#2{\cherule \ethick#1\elength\dimexpr#2/2 \elength\z@}
527
     \global\setbox \tabu@leads=\@box{%
528
         {#3\tabu@therule{#1}{#2}}%
         \left(\frac{\#1}{\#4\times2}\right)
529
         {#3\tabu@therule{#1}{#2}}}%
530
531
     \global\setbox\tabu@leads=\@box to\@elt\tabu@leads{\@ss
```

11.8 The entry inside tabu

\tabu, \endtabu, \longtabu and \endlontabu

\tabu \tabu and \longtabu are the commands of the environments.

\endtabu \endtabu is \endtabular or \endarray in math mode.

```
539 \newcommand*\tabu {\tabu@longfalse
540
                                \end{abular} \en
541
542
                                 \expandafter\let\csname tabu*\endcsname \tabu
                                 \expandafter\def\csname endtabu*\endcsname{\endtabu}%
544
                                 \tabu@spreadfalse \tabu@negcoeffalse \tabu@settarget
545 }% {tabu}
546 \let\tabu@tabular \tabular % <For LyX: some users redefine \tabular...>
547 \expandafter\def\csname tabu*\endcsname{\tabuscantokenstrue \tabu}
548 \newcommand*\longtabu {\tabu@longtrue
                            \ifmmode\PackageError{tabu}{longtabu not allowed in math mode}\fi
550
                            \def\tabu@{\longtable}\def\endlongtabu{\endlongtable}%
551
                           \LTchunksize=\@M
552
                           \expandafter\let\csname tabu*\endcsname \tabu
                           \expandafter\def\csname endlongtabu*\endcsname{\endlongtabu}%
553
554
                            \let\LT@startpbox \tabu@LT@startpbox % \everypar{ array struts }
                            \tabu@spreadfalse \tabu@negcoeffalse \tabu@settarget
556 }% {longtabu}
557 \expandafter\def\csname longtabu*\endcsname{\tabuscantokenstrue \longtabu}
558 \ensuremath{\mbox{\sc bu}} \ensuremath{\mb
                             {longtabu requires the longtable package}\@ehd}
559
```

Setting the tabu target

\tabu@settarget \tabu@begin

The macro sets \tabu@target (a dimen) to the value specified for "tabu to" or "tabu spread".

566 \z@\fi\fi\fi 567 \expandafter\tabu@begin

 $\verb| for with the least of the$

569 \or \expandafter\tabu@to

570 \or \expandafter\tabu@spread 571 \fi

011 /11

572 }% \tabu@sett@rget

 $573 \ensuremath{\mbox{\mbox{1}}} tabu@to to{\ensuremath{\mbox{\mbox{1}}}} tabu@to{\ensuremath{\mbox{1}}} tabu@gettarget}$

 $574 \end{tabu@spread} trabu@spreadtrue\end{tabu@halignto{spread} \tabu@gettarget} \\$

575 \def\tabu@gettarget {\afterassignment\tabu@linegoaltarget \tabu@target }

576 \def\tabu@linegoaltarget {\futurelet\tabu@temp \tabu@linegoalt@rget }

577 \def\tabu@linegoalt@rget {%

```
578
      \ifx \tabu@temp\LNGL@setlinegoal
           \LNGL@setlinegoal \expandafter \@firstoftwo \fi % @gobbles \LNGL@setlinegoal
579
580
      \tabu@begin
581 \ \tabu@linegoalt@rget
582 \leq 14\%
583
       \iftabu@measuring \expandafter\tabu@nestedmeasure \fi
584
       \ifdim \tabu@target=\z@ \let\tabu@halignto \@empty
                                \edef\tabu@halignto{\tabu@halignto\the\tabu@target}%
585
       \fi
586
       \@testopt \tabu@tabu@ \tabu@aligndefault #1\@nil
587
588 }% \tabu@begin
589 \log \det \#3{tabu@tabu@ [#1]#2\enil #3{tabu@setup}
      \def\tabu@align {#1}\def\tabu@savedpream{\NC@find #3}%
      \tabu@ [\tabu@align ]#2{#3\tabu@rewritefirst }%
591
592 }% \tabu@tabu@
593 \def \tabu@nestedmeasure {%}
       \ifodd 1\iftabu@spread \else \ifdim\tabu@target=\z@ \else 0 \fi\fi\relax
594
595
               \tabu@spreadtrue
               \begingroup \iffalse{\fi \ifnum0='}\fi
596
       \else
597
               \toks@{}\def\tabu@stack{b}%
598
               \expandafter\tabu@collectbody\expandafter\tabu@quickrule
599
                                             \expandafter\endgroup
600
       \fi
601 }% \tabu@nestedmeasure
602 \def\tabu@quickrule {\indent\vrule height\z@ depth\z@ width\tabu@target}
```

\tabu@setup \tabu@init \tabu@indent \tabu@init is expanded only when tabu is not nested. In this case, and if \parindent > 0, and if \tabudefaulttarget = \linewidth, the correction of the default target for paragraph indentation is executed (see paragraph indentation).

```
603 \def\tabu@setup{\tabu@alloc@
604
       \ifcase \tabu@nested
605
           \ifmmode \else \iftabu@spread\else \ifdim\tabu@target=\z@
606
               \let\tabu@afterendpar \par
607
           \fi\fi\fi
608
           \def\tabu@aligndefault{c}\tabu@init \tabu@indent
                   % <nested tabu>
609
       \else
           \def\tabu@aligndefault{t}\let\tabudefaulttarget \linewidth
610
611
612
       \let\tabu@thetarget \tabudefaulttarget \let\tabu@restored \@undefined
       \edef\tabu@NC@list{\the\NC@list}\NC@do \tabu@rewritefirst}%
613
614
       \everycr{}\let\@startpbox \tabu@startpbox % for nested tabu inside longtabu...
615
                 \let\@endpbox
                                 \tabu@endpbox
                                                % idem "
                                \tabu@tabarray % idem "
616
                 \let\@tabarray
617
       \tabu@setcleanup \tabu@setreset
618 }% \tabu@setup
619 \def\tabu@init{\tabu@starttimer \tabu@measuringfalse
620
       \edef\tabu@hfuzz {\the\dimexpr\hfuzz+1sp}\global\tabu@footnotes{}%
621
       \let\firsthline
                          \tabu@firsthline
                                             \let\lasthline
                                                                  \tabu@lasthline
622
       \let\firstline
                          \tabu@firstline
                                             \let\lastline
                                                                  \tabu@lastline
623
       \let\hline
                          \tabu@hline
                                              \let\@xhline
                                                                  \tabu@xhline
624
       \let\color
                          \tabu@color
                                             \let\@arstrutbox
                                                                  \tabu@arstrutbox
625
       \iftabu@colortbl\else\let\LT@@hline
                                             \tabu@LT@@hline \fi
626
       \tabu@trivlist
                          %<restore \\=\@normalcr inside lists>
627
       \let\@footnotetext \tabu@footnotetext \let\@xfootnotetext \tabu@xfootnotetext
       \let\@xfootnote
                          \tabu@xfootnote
                                              \let\centering
                                                                  \tabu@centering
629
       \let\raggedright
                          \tabu@raggedright \let\raggedleft
                                                                  \tabu@raggedleft
630
       \let\tabudecimal
                          \tabu@tabudecimal \let\Centering
                                                                  \tabu@Centering
631
       \let\RaggedRight
                          \tabu@RaggedRight \let\RaggedLeft
                                                                  \tabu@RaggedLeft
```

```
632
       \let\justifying
                           \tabu@justifying
                                               \let\rowfont
                                                                     \tabu@rowfont
633
                           \tabu@fbox
                                                                     \tabu@color@b@x
       \let\fbox
                                               \let\color@b@x
634
       \let\tabu@@everycr \everycr
                                               \let\tabu@@everypar \everypar
635
       \let\tabu@prepnext@tokORI \prepnext@tok\let\prepnext@tok \tabu@prepnext@tok
       \let\tabu@multicolumnORI\multicolumn \let\multicolumn
636
                                                                     \tabu@multicolumn
637
       \let\tabu@startpbox \@startpbox
                                              % for nested tabu inside longtabu pfff !!!
638
       \let\tabu@endpbox
                            \@endpbox
                                              % idem
                                                                 11
639
       \let\tabu@tabarray \@tabarray
                                              % idem
                                                      "
                                                            11
640
       \tabu@adl@fix
                          \let\endarray
                                              \tabu@endarray % <fix> colortbl & arydshln (delarray)
       \iftabu@colortbl\CT@everycr\expandafter{\expandafter\iftabu@everyrow \the\CT@everycr \fi}\fi
641
642 }% \tabu@init
643 \def\tabu@indent{% correction for indentation
       \ifdim \parindent>\z@\ifx \linewidth\tabudefaulttarget
645 %
646
       \everypar\expandafter{%
647
           \the\everypar\everypar\expandafter{\the\everypar}%
               \setbox\z@=\lastbox
648
649
               \left(\frac{z}{z}\right) = \left(\frac{z}{z}\right)
                    {\the\dimexpr -\wd\z@+\tabudefaulttarget}\fi
651
               \box\z@}%
       \fi\fi
652
653 }% \tabu@indent
```

\tabu@setcleanup

We have to save locally (in the group of the environment) the current value of the last global assignments to \CT@arc@, \CT@drsc@, \tabu@ls@ etc.

\tabu@cleanup Restoration will be done globally after the box that contains the tabular by \tabu@cleanup.

```
654 \def\tabu@setcleanup {% saves last global assignments
655
       \ifodd 1\ifmmode \else \iftabu@long \else 0\fi\fi\relax
656
           \def\tabu@aftergroupcleanup{%
657
                    \def\tabu@aftergroupcleanup{\aftergroup\tabu@cleanup}}%
658
       \else
           \def\tabu@aftergroupcleanup{%
659
660
                    \aftergroup\aftergroup\tabu@cleanup
661
                    \let\tabu@aftergroupcleanup \relax}%
       \fi
662
663
       \let\tabu@arc@Gsave
                                    \tabu@arc@G
       \let\tabu@arc@G
664
                                    \tabu@arc@L
                                                   % <init>
665
       \let\tabu@drsc@Gsave
                                    \tabu@drsc@G
666
       \let\tabu@drsc@G
                                    \tabu@drsc@L % <init>
667
       \let\tabu@ls@Gsave
                                    \tabu@ls@G
       \let\tabu@ls@G
                                    \tabu@ls@L
668
                                                   % <init>
669
       \let\tabu@rc@Gsave
                                    \tabu@rc@G
670
       \let\tabu@rc@G
                                    \tabu@rc@L
                                                   % <init>
671
       \let\tabu@evr@Gsave
                                    \tabu@evr@G
       \let\tabu@evr@G
                                    \tabu@evr@L
                                                   % <init>
672
673
       \let\tabu@celllalign@save
                                    \tabu@celllalign
       \let\tabu@cellralign@save
                                    \tabu@cellralign
674
675
       \let\tabu@cellleft@save
                                    \tabu@cellleft
676
       \let\tabu@cellright@save
                                    \tabu@cellright
677
       \let\tabu@@celllalign@save
                                    \tabu@@celllalign
678
       \let\tabu@@cellralign@save
                                    \tabu@@cellralign
       \let\tabu@@cellleft@save
                                    \tabu@@cellleft
679
       \let\tabu@@cellright@save
                                    \tabu@@cellright
680
681
       \let\tabu@rowfontreset@save \tabu@rowfontreset
682
       \let\tabu@@rowfontreset@save\tabu@@rowfontreset
683
       \let\tabu@rowfontreset
                                    \@empty
684
       \edef\tabu@alloc@save
                                   {\the\tabu@alloc}%
                                                         restore at \tabu@reset
```

```
685
       \edef\c@taburow@save
                                    {\the\c@taburow}%
       \edef\tabu@naturalX@save
                                    {\the\tabu@naturalX}%
686
687
       \let\tabu@naturalXmin@save
                                    \tabu@naturalXmin
       \let\tabu@naturalXmax@save
                                    \tabu@naturalXmax
688
689
       \let\tabu@mkarstrut@save
                                     \tabu@mkarstrut
690
       \edef\tabu@clarstrut{%
691
           \extrarowheight \the\dimexpr \ht\@arstrutbox-\ht\strutbox \relax
692
           \extrarowdepth \the\dimexpr \dp\@arstrutbox-\dp\strutbox \relax
           \let\noexpand\@arraystretch \@ne \noexpand\tabu@rearstrut}%
693
694 }% \tabu@setcleanup
695 \def\tabu@cleanup {\begingroup
696
       \globaldefs\@ne
                                \tabu@everyrowtrue
697
       \let\tabu@arc@G
                                \tabu@arc@Gsave
       \let\CT@arc@
                                \tabu@arc@G
698
699
       \let\tabu@drsc@G
                                \tabu@drsc@Gsave
700
       \let\CT@drsc@
                                \tabu@drsc@G
701
       \let\tabu@ls@G
                                \tabu@ls@Gsave
702
       \let\tabu@ls@
                                \tabu@ls@G
                                \tabu@rc@Gsave
703
       \let\tabu@rc@G
704
       \let\tabu@rc@
                                \tabu@rc@G
705
       \let\CT@do@color
                                \relax
706
       \let\tabu@evr@G
                                \tabu@evr@Gsave
707
       \let\tabu@celllalign
                                \tabu@celllalign@save
708
       \let\tabu@cellralign
                                \tabu@cellralign@save
       \let\tabu@cellleft
                                \tabu@cellleft@save
709
710
       \let\tabu@cellright
                                \tabu@cellright@save
       \let\tabu@@celllalign
711
                                \tabu@@celllalign@save
                                \tabu@@cellralign@save
712
       \let\tabu@@cellralign
       \let\tabu@@cellleft
                                \tabu@@cellleft@save
713
714
       \let\tabu@@cellright
                                \tabu@@cellright@save
715
       \let\tabu@rowfontreset \tabu@rowfontreset@save
716
       \let\tabu@@rowfontreset \tabu@@rowfontreset@save
717
       \tabu@naturalX
                               =\tabu@naturalX@save
       \let\tabu@naturalXmax
718
                                \tabu@naturalXmax@save
719
       \let\tabu@naturalXmin
                                \tabu@naturalXmin@save
720
       \let\tabu@mkarstrut
                                \tabu@mkarstrut@save
721
       \c@taburow
                               =\c@taburow@save
722
       \ifcase \tabu@nested
                                \tabu@alloc \m@ne\fi
                                % <end of \globaldefs>
723
       \endgroup
724
       \ifcase \tabu@nested
725
           \the\tabu@footnotes \global\tabu@footnotes{}%
726
           \tabu@afterendpar
                                \tabu@elapsedtime
727
       \fi
728
       \tabu@clarstrut
       \everyrow\expandafter
                                {\tabu@evr@G}%
729
730 }% \tabu@cleanup
731 \let\tabu@afterendpar \relax
```

At the beginning of each trial, we have to restore the current value that were active at the entry in the tabu environment (for they could have been globally overwritten inside the tabular).

The same must occur when using \usetabu as a preamble. Values are restored locally inside the tabu box.

\tabu@setreset defines \tabu@reset to be expanded at the beginning of each trial and when \usetabu is used.

```
732 \ensuremath{\mbox{def}\tabu@setreset} {%
733
        \edef\tabu@savedparams {%
                                                \relax for \tabu@message@save
734
             \ifmmode \col@sep \the\arraycolsep
735
             \else
                       \col@sep \the\tabcolsep \fi
                                                           \relax
```

```
736
           \arrayrulewidth
                             \the\arrayrulewidth
                                                    \relax
           \doublerulesep
                             \the\doublerulesep
737
                                                    \relax
738
           \extratabsurround \the\extratabsurround \relax
           \extrarowheight
                             \the\extrarowheight
                                                    \relax
739
740
           \extrarowdepth
                             \the\extrarowdepth
                                                    \relax
741
           \abovetabulinesep \the\abovetabulinesep \relax
742
           \belowtabulinesep \the\belowtabulinesep \relax
743
           \def\noexpand\arraystretch{\arraystretch}%
           \ifdefined\minrowclearance \minrowclearance\the\minrowclearance\relax\fi}%
744
       \begingroup
745
           \label{thm:constraints} $$ => only for \avetabu / \avetabu .
746
747
           \ifx \tabu@arc@L\relax \else \tabu@setsave \tabu@arc@L \fi
           \ifx \tabu@drsc@L\relax \else \tabu@setsave \tabu@drsc@L \fi
748
749
           \tabu@setsave \tabu@ls@L
                                          \tabu@setsave \tabu@evr@L
750
           \expandafter \endgroup \expandafter
               751
                   \let\tabu@arc@G \tabu@arc@L
752
753
                   \let\tabu@drsc@G \tabu@drsc@L
                   \let\tabu@ls@G
                                    \tabu@ls@L
                   \let\tabu@rc@G
                                    \tabu@rc@L
755
                   \let\tabu@evr@G \tabu@evr@L}%
756
757
       \def\tabu@reset{\tabu@savedparams
758
           \tabu@everyrowtrue \c@taburow \z@
759
           \let\CT@arc@
                                \tabu@arc@L
           \let\CT@drsc@
                               \tabu@drsc@L
760
761
           \let\tabu@ls@
                               \tabu@ls@L
                               \tabu@rc@L
762
           \let\tabu@rc@
763
           \global\tabu@alloc \tabu@alloc@save
           \everyrow\expandafter{\tabu@evr@L}}%
765 }% \tabu@reset
766 \def\tabu@setsave \#1{\exp{\text{andafter}}\
767 \long\def\tabu@sets@ve \#1\ensuremath{\mathchar} #1\\\ensuremath{\mathchar} #1\\\ensuremath{\mathchar} \def\#2\\\ensuremath{\mathchar} +1\\\\ensuremath{\mathchar}
```

11.9 The rewriting process: inside the "\@mkpream group"

New column types and private (new) column types

\tabu@newcolumntype A helper macro to create new column types for tabu.

The column types are not appended to \NC@list in order to keep them local to tabu.

```
768 \def\tabu@newcolumntype #1{%
769
       \expandafter\tabu@new@columntype
770
           \csname NC@find@\string#1\expandafter\endcsname
           \csname NC@rewrite@\string#1\endcsname
771
772
           {#1}%
773 }% \tabu@newcolumntype
774 \def\tabu@new@columntype #1#2#3{%
       \def#1##1#3{\NC@{##1}}%
775
       \let#2\relax \newcommand*#2%
777 }% \tabu@new@columntype
```

group of tabu. $\mathsf{T}_{\aleph}b \subset [\text{rev.}2.8 \text{ release}] \odot 2010-$

```
778 \def\tabu@privatecolumntype #1{%
       \expandafter\tabu@private@columntype
779
           \csname NC@find@\string#1\expandafter\endcsname
780
           \csname NC@rewrite@\string#1\expandafter\endcsname
781
782
           \csname tabu@NC@find@\string#1\expandafter\endcsname
783
           \csname tabu@NC@rewrite@\string#1\endcsname
```

```
784 {#1}%
785}% \tabu@privatecolumntype
786 \def\tabu@private@columntype#1#2#3#4{%
787 \g@addto@macro\tabu@privatecolumns{\let#1#3\let#2#4}%
788 \tabu@new@columntype#3#4%
789}% \tabu@private@columntype
790 \let\tabu@privatecolumns \@empty
```

High priority columns

802 \def\tabu@vlinearg #1{%

\tabucolumn \tabucolumn puts a user-defined column in high priority in the tabu rewriting process.

```
791 \newcommand*\tabucolumn [1]{\expandafter \def \expandafter
792 \tabu@highprioritycolumns\expandafter{\tabu@highprioritycolumns
793 \NC@do #1}}%
794 \let\tabu@highprioritycolumns \@empty
```

Rewriting vertical lines and leaders

I (private column type) This is the rewrite macro for the I column type inside tabu and longtabu.

Vertical lines are *simply rewritten* as special! columns.

```
795 \tabu@privatecolumntype |{\tabu@rewritevline}
796 \newcommand*\tabu@rewritevline[1][]{\tabu@vlinearg{#1}%
797 \expandafter \NC@find \tabu@rewritten}
```

\tabu@lines

The | token for vertical lines may have a special catcode. array.sty makes the test with \if and therefore, it is catcode insensitiv. Here, we use \scantokens and check if | is not an other character.

```
798 \def\tabu@lines #1{%
799 \ifx|#1\else \tabu@privatecolumntype #1{\tabu@rewritevline}\fi
800 \NC@list\expandafter{\the\NC@list \NC@do #1}%
801 }% \tabu@lines@
```

\tabu@vlinearg

The macro that parses the optional argument of | vertical lines...

```
803
      \ifx\\#1\\\def\tabu@thestyle {\tabu@ls@}%
804
      \else\tabu@getline {#1}%
805
      \def\tabu@rewritten ##1{\def\tabu@rewritten{!{##1\tabu@thevline}}%
806
807
      }\expandafter\tabu@rewritten\expandafter{\tabu@thestyle}%
      \expandafter \tabu@keepls \tabu@thestyle \@nil
809 }% \tabu@vlinearg
810 \def\tabu@keepls #1\@nil{%
      \ifcat $\@cdr #1\@nil $%
811
812
      \ifx \relax#1\else
      \ifx \tabu@ls@#1\else
813
814
          \let#1\relax
          \xdef\tabu@mkpreambuffer{\tabu@mkpreambuffer
                  \tabu@savels\noexpand#1}\fi\fi\fi
817 }% \tabu@keepls
818 \def\tabu@thevline {\begingroup
819
      \ifdefined\tabu@leaders
820
          \setbox\@tempboxa=\vtop to\dimexpr
                        \ht\@arstrutbox+\dp\@arstrutbox{{\tabu@thevleaders}}%
821
822
          823
          \box\@tempboxa
      \else
824
                  \tabu@thevrule
825
826
      \fi
                      \endgroup
827 }% \tabu@thevline
```

```
828 \def\tabu@savels #1{%
829 \expandafter\let\csname\string#1\endcsname #1%
830 \expandafter\def\expandafter\tabu@reset\expandafter{\tabu@reset}
831 \tabu@resetls#1}}%
832 \def\tabu@resetls #1{\expandafter\let\expandafter#1\csname\string#1\endcsname}%
```

Vertical lines and leaders in the \multicolumn preamble

\tabu@rewritemulticolumn A special rewrite to allow [...] in \multicolumn preamble inside tabu environment.

As long as \multicolumn begins with \omit (via \multispan) special care has to be taken: everything shall be purely expandable until \omit.

\multicolumn is not an environment: no group is opened apart the \@mkpream group. We open a semi simple group for \multicolumn when inside tabu, in order for the setup to be local (in case a user would try to embed a tabular inside the argument of \multicolumn...)

```
833 \tabu@newcolumntype \tabu@rewritemulticolumn{%
       \aftergroup \tabu@endrewritemulticolumn % after \@mkpream group
835
       \NC@list{\NC@do *}\tabu@textbar \tabu@lines
       \tabu@savedecl
836
       \tabu@privatecolumns
837
838
       \NC@list\expandafter{\the\expandafter\NC@list \tabu@NC@list}%
       \let\tabu@savels \relax
840
       \NC@find
841 }% \tabu@rewritemulticolumn
842 \def\tabu@endrewritemulticolumn{\gdef\tabu@mkpreambuffer{}\endgroup}
843 \def\tabu@multicolumn{\tabu@ifenvir \tabu@multic@lumn \tabu@multicolumnORI}
844 \long\def\tabu@multic@lumn #1#2#3{\multispan{#1}\begingroup
       \tabu@everyrowtrue
       \NC@list{\NC@do \tabu@rewritemulticolumn}%
846
       \expandafter\@gobbletwo % gobbles \multispan{#1}
847
848
            \tabu@multicolumnORI{#1}{\tabu@rewritemulticolumn #2}%
                   {\iftabuscantokens \tabu@rescan \else \expandafter\@firstofone \fi
849
850
851 }% \tabu@multic@lumn
```

Rewriting tabu X columns

x (private column type) This is the rewrite macro for tabu X columns. Such a column has an optional argument: the width coefficient for the tabu X column whose default value is 1, and may be some alignments parameters. The coefficient is used in the expression: p{\dimexpr \langle coef} \tabucolX }

```
853 \neq 141{1{1{1}}}
854 \def\tabu@siunitx
                   #1{\@ifnextchar \bgroup
855
                    {\tabu@rewriteX@Ss{#1}}
856
                     {\tabu@nosiunitx{#1}}}
857 \def \tabu@rewriteX@Ss #1#2{\dtemptokena{}%}
858
      \@defaultunits \let\tabu@temp =#2\relax\@nnil
859
      \ifodd 1\ifx S\tabu@temp \else \ifx s\tabu@temp \else 0 \fi\fi
         \def\NC@find{\def\NC@find >####1###2<####3\relax{#1 {####1}{####3}%
860
             }\expandafter\NC@find \the\@temptokena \relax
862
         }\expandafter\NC@rewrite@S \@gobble #2\relax
      \else \tabu@siunitxerror
863
864
      \expandafter \NC@find \tabu@rewritten
865
866 }% \tabu@rewriteX@Ss
867 \def\tabu@siunitxerror {\PackageError{tabu}{Not a S nor s column !
         \MessageBreak X column can only embed siunitx S or s columns}\@ehd
868
869 }% \tabu@siunitxerror
```

\tabu@rewriteX

This macro is expanded by during the rewriting process in case a X column is found.

\tabu@Xsum (a dimen) stores the sum of the (absolute) width coefficients.

For the first X column found in the preamble, a special setup occurs:

- if the default target is used (no target specified or tabu spread with X columns), the target: \tabu@target is set to the default, with a message in the .log file.
- \Chalignto is \let to \relax to avoid its expansion in \xdef \Congreamble just after \Comkpream. Indeed as long as we have to measure the natural width of the tabular, \Chalign must be empty for trial steps.
- The rest of the setup is made \aftergroup (ie.after \xdef \@preamble which occurs inside a group) by \tabu@prep@TRIAL.

```
870 \def \tabu@rewriteX #1#2#3{\tabu@Xarg {#1}{#2}{#3}%
871
       \iftabu@measuring
       \else \tabu@measuringtrue % first X column found in the preamble
872
           \let\@halignto \relax
                                   \let\tabu@halignto \relax
873
874
           \iftabu@spread \tabu@spreadtarget \tabu@target \z@
           \else
                          \tabu@spreadtarget \z@ \fi
875
876
           \ifdim \tabu@target=\z@
877
                   \setlength\tabu@target \tabu@thetarget
                   \tabu@message{\tabu@message@defaulttarget}%
878
879
                   \tabu@message{\tabu@message@target}\fi
           \else
880
       \fi
881 }% \tabu@rewriteX
```

\tabu@rewriteXrestore This macro replaces \tabu@rewriteX in the case of \usetabu.

\tabu@Xarg \tabu@Xparse A tedious (and fastidious) macro to parse the optional argument of X columns. The aim is to built \tabu@rewritten which expands to the column specification:

>{alignment} p or m or b {\dimexpr coef \tabucolX \relax }

After that array.sty make it easy: \expandafter \NC@find \tabu@rewritten

```
884 \def\tabu@Xarg #1#2#3{%
      \advance\tabu@Xcol \@ne
885
                                   \let\tabu@Xlcr
                                                  \@emptv
886
      \let\tabu@Xdisp
                        \@empty
                                   \let\tabu@Xmath \@empty
887
       \ifx\\#1\\%
                     <shortcut when no option>
            \def\tabu@rewritten{p}\tabucolX \p@
                                                       % <default coef = 1>
888
889
       \else
            \let\tabu@rewritten \@empty
                                         890
            \tabu@Xparse {}#1\relax
891
       \tabu@Xrewritten{#2}{#3}%
893
894 }% \tabu@Xarg
895 \def\tabu@Xparse #1{\futurelet\@let@token \tabu@Xtest}
896 \expandafter\def\expandafter\tabu@Xparsespace\\tabu@Xparse{}}
897 \def\tabu@Xtest{%
898
      \ifcase \ifx \relax\@let@token \z@ \else
              \if ,\@let@token \m@ne\else
899
900
              \if p\@let@token 1\else
              \if m\@let@token 2\else
901
902
              \if b\@let@token 3\else
              \if l\@let@token 4\else
903
              \if c\@let@token 5\else
904
905
              \if r\@let@token 6\else
906
              \if j\@let@token 7\else
              \if L\@let@token 8\else
907
```

```
908
                                                                                  \if C\@let@token 9\else
                                                909
                                                                                   \if R\@let@token 10\else
                                                910
                                                                                   \if J\@let@token 11\else
                                                911
                                                                                   \ifx \@sptoken\@let@token 12\else
                                                                                   \if .\@let@token 13\else
                                                912
                                                913
                                                                                  \if -\@let@token 13\else
                                                914
                                                                                   \ifcat $\@let@token 14\else
                                                915
                                                                                  916
                                                               \or \tabu@Xtype {p}%
                                                917
                                                               \or \tabu@Xtype {m}%
                                                               \or \tabu@Xtype {b}%
                                                918
                                                919
                                                               \or \tabu@Xalign \raggedright\relax
                                                920
                                                               \or \tabu@Xalign \centering\relax
                                                921
                                                               \or \tabu@Xalign \raggedleft\relax
                                                               \or \tabu@Xalign \tabu@justify\relax
                                                922
                                                               \or \tabu@Xalign \RaggedRight\raggedright
                                                923
                                                924
                                                               \or \tabu@Xalign \Centering\centering
                                                925
                                                               \or \tabu@Xalign \RaggedLeft\raggedleft
                                                               \or \tabu@Xalign \justifying\tabu@justify
                                                927
                                                               \or \expandafter \tabu@Xparsespace
                                                               \or \expandafter \tabu@Xcoef
                                                928
                                                               \or \expandafter \tabu@Xm@th
                                                929
                                                930
                                                               \or \tabu@Xcoef{}%
                                                931
                                                               \else\expandafter \tabu@Xparse
                                                932
                                                               \fi
                                                933 }% \tabu@Xtest
                                                934 \def \align #1#2{%}
                                                                 \ifx \tabu@Xlcr\@empty \else \PackageWarning{tabu}
                                                                          {Duplicate horizontal alignment specification}\fi
                                                936
                                                937
                                                                 \ifdefined#1\def\tabu@Xlcr{#1}\let#1\relax
                                                938
                                                                                               \def\tabu@Xlcr{#2}\let#2\relax\fi
                                                939
                                                                 \expandafter\tabu@Xparse
                                                940 \ \tabu@Xalign
                                                941 \def\tabu@Xtype #1{%
                                                                 \ifx \tabu@rewritten\@empty \else \PackageWarning{tabu}
                                                942
                                                                                      {Duplicate vertical alignment specification}\fi
                                                                 \def\tabu@rewritten{#1}\expandafter\tabu@Xparse
                                                945 }% \tabu@Xtype
                                                946 \ensuremath{\texttt{4}} \ensuremath{\texttt{2}} \ensuremath{\texttt{4}} \ensur
                                                                 \afterassignment\tabu@Xc@ef \tabu@cnt\number\if-#10\fi
                                                948 }% \tabu@Xcoef
                                                949 \def\tabu@Xc@ef{\advance\tabucolX \tabu@temp\the\tabu@cnt\p@
                                                                 \tabu@Xparse{}%
                                                951}% \tabu@Xc@ef
```

```
The properties of the propert
```

```
964 \edef\tabu@temp{{\the\tabu@Xcol}{\tabu@strippt\tabucolX}}%
965 \edef\tabu@Xcoefs{\tabu@Xcoefs \tabu@ \tabu@temp}%
966 \edef\tabu@rewritten ##1##2{\def\noexpand\tabu@rewritten{%
967 >{\tabu@Xlcr \ifx$\tabu@Xmath$\tabu@Xdisp\fi ##1}%
968 \tabu@rewritten {\tabu@hsize \tabu@temp}%
969 <{##2\ifx$\tabu@Xmath$\fi}}%
970 }\tabu@rewritten</pre>
971 }% \tabu@Xrewritten
```

\tabu@hsize \tabu@hsize {X column number}{X column width coefficient}

Depending on the sign of the coefficient, and of the stored value for the natural width of the column the X cell belongs to, \tabu@hsize returns the wanted width for the par-box that contains the cell content.

```
972 \def\tabu@hsize #1#2{%

973 \ifdim #2\p@<\z@

974 \ifdim \tabucolX=\maxdimen \tabu@wd{#1}\else

975 \ifdim \tabu@wd{#1}<-#2\tabucolX \tabu@wd{#1}\else -#2\tabucolX\fi

976 \fi

977 \else #2\tabucolX

978 \fi

979 }% \tabu@hsize
```

Rewritting \usetabu and \preamble

The rewritting process is very simple, when all the job has been done cleverly at the time of \savetabu!! The \savetabu macro is a bit more complex...

\usetabu (private column type) \usetabu is defined as a tabu new column type: loaded only inside the \@mkpream group inside the tabu environment.

```
980 \tabu@privatecolumntype \usetabu [1]{%

981 \ifx\\#1\\tabu@saveerr{}\else

982 \@ifundefined{tabu@saved@\string#1}

983 {\tabu@saveerr{#1}}

984 {\let\tabu@rewriteX \tabu@rewriteXrestore

985 \csname tabu@saved@\string#1\expandafter\endcsname\expandafter\@ne}%

986 \fi

987 }% \NC@rewrite@\usetabu
```

\preamble (private column type) \preamble is defined as a tabu new column type: loaded only inside the \@mkpream group inside the tabu environment.

```
988 \tabu@privatecolumntype \preamble [1] {\\
989 \ifx\\#1\\tabu@saveerr{}\else
990 \@ifundefined{tabu@saved@\string#1}
991 {\tabu@saveerr{#1}}
992 {\csname tabu@saved@\string#1\expandafter\endcsname\expandafter\z@}\%
993 \fi
994 }\% \NC@rewrite@\preamble
```

Controlling the rewritting process

This new column type is not really a column type! It is always added to a tabu preamble in order to do some setup before any other column is rewritten by \@mkpream.

Thus, \NC@list is simply set to {\NC@do \tabu@rewritefirst }. The rewritting of \tabu@rewritefirst will restore the original list \NC@list.

This "column type" sets:

• \tabu@select to be expanded \aftergroup (after the closing of the \@mkpream group. All the thick is there: all information collected during the rewritting of X columns (and vertical lines or leaders) can be reinjected into the group below the \@mkpream group, by the mean of the

\tabu@mkpreambuffer (globally defined).

- The private columns types are loaded by \tabu@rewritefirst: they will be rewritten afterwards, during the rewritting loop. This way, X column definition for tabu are only available during the rewritting process of the tabu preamble, making it possible (and easy) to embed a tabularx inside a cell of a tabu.
- \save@decl is modified inside the \@mkpream group, if tabu is in text mode.

```
995 \tabu@newcolumntype \tabu@rewritefirst{%
        \iftabu@long
                        \aftergroup \tabu@longpream % <the whole implementation is here !>
996
        \else
                        \aftergroup \tabu@pream
997
        \fi
998
999
        \let\tabu@
                             \relax
                                         \let\tabu@hsize
                                                              \relax
1000
        \let\tabu@Xcoefs
                             \@empty
                                         \let\tabu@savels
                                                              \relax
1001
        \tabu@Xcol
                             \z@
                                         \tabu@cnt
                                                              \tw@
1002
        \gdef\tabu@mkpreambuffer{\tabu@{}}\tabu@measuringfalse
1003
        \global\setbox\@arstrutbox \box\@arstrutbox
1004
        \NC@list{\NC@do *}\tabu@textbar \tabu@lines
        \NC@list\expandafter{\the\NC@list \NC@do X}%
1005
        \iftabu@siunitx
                             % <siunitx S and s columns>
1006
                \NC@list\expandafter{\the\NC@list \NC@do S\NC@do s}\fi
1007
1008
        \NC@list\expandafter{\the\expandafter\NC@list \tabu@highprioritycolumns}%
1009
        \expandafter\def\expandafter\tabu@NC@list\expandafter{%
1010
                        \the\expandafter\NC@list \tabu@NC@list}%
                                                                      % * | X S <original>
        \NC@list\expandafter{\expandafter \NC@do \expandafter\usetabu
1011
1012
                              \expandafter \NC@do \expandafter\preamble
1013
                              \the\NC@list \NC@do \tabu@rewritemiddle
                                           \NC@do \tabu@rewritelast}%
1014
1015
        \tabu@savedecl
1016
        \tabu@privatecolumns
1017
        \edef\tabu@prev{\the\@temptokena}\NC@find \tabu@rewritemiddle
1018 }% NC@rewrite@\tabu@rewritefirst
```

\tabu@rewritemiddleThis new column type is rewritten after X columns, because it is declared by when the column \tabu@rewritelast \tabu@rewritefirst is actually rewritten. In the case where \tabu@target is > 0 (either because of "tabu to" or "tabu spread" has been called) and if there is no X column, then @{\extracolsep \@flushglue } is added at the beginning of the preamble.

To avoid duplicate margin in the tabu we have to test the next token in the preamble. If the next token is | or ! then no margin must be added and <code>@{\extracolsep \@flushglue }</code> can be inserted at the beginning of the preamble.

Otherwise, we must insert !{\extracolsep \@flushglue } in order to keep the margin.

\tabu@rewritelast column type is loaded by \tabu@rewritefirst column type, only inside the \@mkpream group inside the tabu environment.

```
1019 \tabu@newcolumntype \tabu@rewritemiddle{%
        \edef\tabu@temp{\the\@temptokena}\NC@find \tabu@rewritelast
1020
1021 }% \NC@rewrite@\tabu@rewritemiddle
1022 \tabu@newcolumntype \tabu@rewritelast{%
1023
        \ifx \tabu@temp\tabu@prev \advance\tabu@cnt \m@ne
1024
                \NC@list\expandafter{\tabu@NC@list \NC@do \tabu@rewritemiddle
1025
                                                    \NC@do \tabu@rewritelast}%
1026
        \else \let\tabu@prev\tabu@temp
1027
        \fi
        \ifcase \tabu@cnt
1028
                            \expandafter\tabu@endrewrite
                             \expandafter\NC@find \expandafter\tabu@rewritemiddle
1029
        \else
1030
        \fi
1031 }% \NC@rewrite@\tabu@rewritelast
```

The end of the rewritting process: determining the tabu strategy

\tabu@endrewrite Determines the strategy to be executed \aftergroup (at the closing of the \@mkpream group):

- 0) There is no real strategy: tabu behaves like tabular, there no X column, and no need to measure the vertical dimensions of the cells (no dynamic spacing, no vertical leader). In case a target has been given to tabu, it behaves like tabular* and a infinite stretchability is given to the column inter-space. This is done (if required) by \tabu@extracolsep.
- 1) Measuring natural width of some (or all) columns is compulsory for tabu spread of X columns with negativ coefficients. Thereafter, the strategy nr 2 will bring into play.
- 2) Measuring the natural width is not necessary, or has been done before. But tabu contains X columns and trials have to be performed to reach the desired target, adjusting the \tabucolX dimension accordingly. Then, the strategy nr 3 may bring into play, if vertical measure is required.
- 3) Vertical measure of the cells is required, for vertical spacing adjustment or vertical leaders. This step can be done only if the width are known.
- > 3 The tabu is finished and ready to be printed!!

```
1032 \def \tabu@endrewrite {%}
        \let\tabu@temp \NC@find
1033
1034
        \ifx \@arrayright\relax \let\@arrayright \@empty
        \count@=%
1035
            \ifx \@finalstrut\tabu@finalstrut \z@ % outer in mode 0 print
1036
1037
                 \iftabu@measuring
1038
                    \xdef\tabu@mkpreambuffer{\tabu@mkpreambuffer
1039
                         \tabu@target
                                              \csname tabu@\the\tabu@nested.T\endcsname
1040
                         \tabucolX
                                              \csname tabu@\the\tabu@nested.X\endcsname
1041
                         \edef\@halignto {\ifx\@arrayright\@empty to\tabu@target\fi}}%
                 \fi
1042
            \else\iftabu@measuring
                                                    % X columns
1043
                                           4
1044
                    \xdef\tabu@mkpreambuffer{\tabu@{\tabu@mkpreambuffer
1045
                         \tabu@target
                                              \the\tabu@target
1046
                         \tabu@spreadtarget \the\tabu@spreadtarget}%
1047
                         \def\noexpand\tabu@Xcoefs{\tabu@Xcoefs}%
                         \edef\tabu@halignto{\ifx \@arrayright\@empty to\tabu@target\fi}}%
1048
                      \let\tabu@Xcoefs \relax
1049
                 \else\ifcase\tabu@nested \thr@@ % outer, no X
1050
1051
                                            \global\let\tabu@afterendpar \relax
1052
                      \else
                                                    % inner, no X, outer in mode 1 or 2
                      \fi
1053
                      \ifdefined\tabu@usetabu
1054
1055
                       \else \ifdim\tabu@target=\z@
                       \else \let\tabu@temp \tabu@extracolsep
1056
1057
                      \fi\fi
                 \fi
1058
1059
            \fi
1060
        \xdef\tabu@mkpreambuffer{\count@ \tabu@mkpreambuffer}%
1061
        \tabu@temp
1062 \}% \tabu@endrewrite
```

 $\mathsf{L}_{\mathsf{N}}b \subset [\mathrm{rev.2.8\ release}] \otimes 2010 - 2010 = 0.000$

Inserts \extracolsep \@flushglue in front of the preamble, unless another value for \extracolsep has been specified.

\@flushglue is Opt plus 1fil.

```
1063 \def\tabu@extracolsep{\@defaultunits \expandafter\let
1064 \expandafter\tabu@temp \expandafter=\the\@temptokena \relax\@nnil
1065 \ifx \tabu@temp\@sptoken
1066 \expandafter\tabu@gobblespace \expandafter\tabu@extracolsep
```

```
1067
        \else
1068
             \edef\tabu@temp{\noexpand\NC@find
1069
                 \if |\noexpand\tabu@temp
                                                     @ 8
                 \else\if !\noexpand\tabu@temp
1070
                                                     @ 8
                                                     ! 응
1071
                 \else
1072
                 \fi\fi
1073
                 {\noexpand\extracolsep\noexpand\@flushglue}}%
1074
        \fi
1075
        \tabu@temp
1076 }% \tabu@extrac@lsep
```

11.10 Implementing the strategy at the exit of the \@mkpream group

\tabu@select

\tabu@pream

Triggered \aftergroup by the rewritting of \tabu@rewritefirst.

The \tabu@mkpreambuffer macro is expanded twice: first it injects \count@ (the strategy number) and \tabu@nbcols, and redefines itself.

Second – and only if measurements are necessary – it expands into the *trials group* to inject \tabu@Xcoefs (the coefficients of X columns), \tabu@Xsum (the sum of the absolute coefficients), \tabu@target, \tabu@spreadtarget, and \tabu@vertical, which is the number by which one have to increment the strategy number after step 2 (either 1: then a last measure is done for the vertical dimensions, or 255 then the strategy number is > 3 and \tabu@strategy orders to finish.)

\tabu@longpream

This is the long version for longtabu: the material to collect until \@preamble is different!

```
1077 \long\def\tabu@pream #1\@preamble {%
1078
        \let\tabu@ \tabu@@ \tabu@mkpreambuffer
                                                     \tabu@aftergroupcleanup
1079
        \NC@list\expandafter {\tabu@NC@list}%
                                                  in case of nesting...
1080
        \ifdefined\tabu@usetabu \tabu@usetabu \tabu@target \z@ \fi
        \let\tabu@savedpreamble \@preamble
1081
1082
        \global\let\tabu@elapsedtime \relax
1083
        \tabu@thebody ={#1\tabu@aftergroupcleanup}%
        \tabu@thebody =\expandafter{\the\expandafter\tabu@thebody
1084
1085
                                                     \@preamble}%
        \edef\tabuthepreamble {\the\tabu@thebody}% ( no @ allowed for \scantokens )
1086
1087
        \tabu@select
1088 }% \tabu@pream
1089 \long\def\tabu@longpream #1\LT@bchunk #2\LT@bchunk{%
        \let\tabu@ \tabu@@ \tabu@mkpreambuffer
1090
                                                     \tabu@aftergroupcleanup
1091
        \NC@list\expandafter {\tabu@NC@list}%
                                                 in case of nesting ...
       \let\tabu@savedpreamble \@preamble
1092
1093
        \global\let\tabu@elapsedtime \relax
1094
        \tabu@thebody ={#1\LT@bchunk #2\tabu@aftergroupcleanup \LT@bchunk}%
1095
        \edef\tabuthepreamble {\the\tabu@thebody}% ( no @ allowed for \scantokens )
1096
        \tabu@select
1097 }% \tabu@longpream
```

\tabu@select

Here we check if trials are required or not: depending on the value of \count@ (set at \tabu@endrewrite, and injected here by \tabu@mkpreambuffer), on \iftabu@measuring (nested trials).

When trials are required, \tabu@select give control to \tabu@setstrategy (to prepare the neutralisation of commands, save counters etc).

When trials are not required, we just have to expand **\tabuthepreamble**, after having set up the **\everyrow** stuff properly (for vertical adjustment or vertical measure, if needed).

```
1098 \def\tabu@select {%
1099 \ifnum\tabu@nested>\z@ \tabuscantokensfalse \fi
1100 \ifnum \count@=\@ne \iftabu@measuring \count@=\tw@ \fi\fi
1101 \ifcase \count@
```

```
1102
            \global\let\tabu@elapsedtime \relax
1103
            \tabu@setevervcr
1104
            \expandafter \tabuthepreamble
                                                  % vertical adjustment (inheritated from outer)
                 \$ exit in vertical measure + struts per cell because no X and outer in mode 3
1105
            \tabu@evr{\tabu@verticalinit}\tabu@celllalign@def{\tabu@verticalmeasure}%
1106
1107
            \def\tabu@cellralign{\tabu@verticalspacing}%
1108
            \tabu@seteverycr
            \expandafter \tabuthepreamble
1109
                                         % exit without measure because no X and outer in mode 4
1110
        \or
            \tabu@evr{}\tabu@celllalign@def{}\let\tabu@cellralign \@empty
1111
1112
            \tabu@seteverycr
1113
            \expandafter \tabuthepreamble
                                                  % needs trials
1114
        \else
1115
            \tabu@evr{}\tabu@celllalign@def{}\let\tabu@cellralign \@empty
1116
            \tabu@savecounters
            \expandafter \tabu@setstrategy
1117
        \fi
1118
1119 }% \tabu@select
1120 \def\tabu@@ {\gdef\tabu@mkpreambuffer}
```

General setup for trials: neutralisation of \write etc.

 $\ifdim #2\p@>\z@ #2\p@ \advance\dimen@ #2\p@$

\tabu@setstrategy

This is the general setup for trials: the tabu will be expanded more than once, thus some protections are set: the value of global counters are saved, footnotes have a special setup, \hbadness and \hfuzz are neutralised etc.

The initial value for \tabucolX is computed with the coefficients stored into \tabu@Wvoefs: \tabu@ \coef1} \tabu@ \coef2} \tabu@ \coef3} etc.

is very suitable for loops on the column width coefficients (without the need of \@for or whatsoever).

```
1121 \def\tabu@setstrategy {\begingroup % <trials group>
1122
       \tabu@trialh@@k
                          \tabu@cnt
                                       \z@ % number of trials
       \hbadness
1123
                          / am
                                       \let\hbadness
                                                             \@tempcnta
1124
       \hfuzz
                          \maxdimen
                                       \let\hfuzz
                                                             \@tempdima
       \let\write
1125
                          \tabu@nowrite\let\GenericError
                                                             \tabu@GenericError
1126
       \let\savetabu
                          \@gobble
                                       \let\tabudefaulttarget \linewidth
1127
       \let\@footnotetext \@gobble
                                       \let\@xfootnote
                                                             \tabu@xfootnote
                          \tabu@nocolor\let\rowcolor
                                                             \tabu@norowcolor
1128
       \let\color
1129
       \let\tabu@aftergroupcleanup \relax % only after the last trial
1130
       \tabu@mkpreambuffer
       \ifnum \count@>\thr@@ \let\@halignto \@empty \tabucolX@init
1131
1132
                             \def\tabu@lasttry{\m@ne\p@}\fi
1133
       \begingroup \iffalse{\fi \ifnum0='}\fi
           1134
1135
                                     \tabu@collectbody \tabu@strategy %
1136 }% \tabu@setstrategy
1137 \def\tabu@savecounters{%
1138
       \def\@elt ##1{\csname c@##1\endcsname\the\csname c@##1\endcsname}%
       \edef\tabu@clckpt {\begingroup \globaldefs=\@ne \cl@@ckpt \endgroup}\let\@elt \relax
1139
1140 }% \tabu@savecounters
1141 \def\tabucolX@init {% \tabucolX <= \tabu@target / (sum coefs > 0)
       \dimen@ \z@ \tabu@Xsum \z@ \tabucolX \z@ \let\tabu@ \tabu@Xinit \tabu@Xcoefs
       \ifdim \dimen@>\z@
1143
1144
           \@tempdima \dimexpr \tabu@target *\p@/\dimen@ + \tabu@hfuzz\relax
1145
           \ifdim \tabucolX<\@tempdima \tabucolX \@tempdima \fi
       \fi
1146
1147 }% \tabucolX@init
1148 \def\tabu@Xinit #1#2{\tabu@Xcol #1 \advance \tabu@Xsum
```

```
1150
        \else
                         -#2\p@ \tabu@negcoeftrue
1151
                                  \@tempdima \dimexpr \tabu@target*\p@/\dimexpr-#2\p@\relax \relax
1152
                                 \ifdim \tabucolX<\@tempdima \tabucolX \@tempdima \fi
1153
                                  \tabu@wddef{#1}{Opt}%
        \fi
1154
1155 }% \tabu@Xinit
```

Collecting the tabu body

The macro collect the stuff inside \@array: depending on the global vertical alignment parameter for the whole tabular, the tabular is built inside a \vbox, \vtop or \vcenter (the default – unless tabu is nested).

At this time, we define \tabu@trial (which inherits from the \vbox, \vtop or \vcenter) and \tabu@Xfinish as well.

\tabu@collectbody

The mechanism is the same as AMS-\collect@body (also defined in environ.sty). The content of the \tabu@endofcollect tabular is captured inside \toks@, expanded by \tabu@trial.

```
1156 \long\def\tabu@collectbody #1#2\end #3{%
1157
       \edef\tabu@stack{\tabu@pushbegins #2\begin\end\expandafter\@gobble\tabu@stack}%
1158
       \ifx \tabu@stack\@empty
1159
            \toks@\expandafter{\expandafter\tabu@thebody\expandafter{\the\toks@ #2}%
1160
                    \def\tabu@end@envir{\end{#3}}%
1161
                    \iftabuscantokens
                        \iftabu@long \def\tabu@endenvir {\end{#3}\tabu@gobbleX}%
1162
1163
                                     \def\tabu@endenvir {\let\endarray \@empty
                                                         \end{#3}\tabu@gobbleX}%
1164
1165
                        \fi
1166
                    \else
                                    \let\tabu@collectbody \tabu@endofcollect
1167
       \else\def\tabu@temp{#3}%
1168
1169
            \ifx \tabu@temp\@empty \toks@\expandafter{\the\toks@ #2\end }%
            \else \ifx\tabu@temp\tabu@@spxiii \toks@\expandafter{\the\toks@ #2\end #3}%
1170
1171
            \else \ifx\tabu@temp\tabu@X \toks@\expandafter{\the\toks@ #2\end #3}%
            \end{#3}}% $$ \end{#3}}% $$ \end{#3}}% $$
1172
            \fi\fi\fi
1173
1174
       \fi
1175
       \tabu@collectbody{#1}%
1176 }% \tabu@collectbody
1177 \long\def\tabu@pushbegins#1\begin#2{\ifx\end#2\else b\expandafter\tabu@pushbegins\fi}%
1178 \def\tabu@endofcollect #1{\imum0=`{}}fi
                              \verb|\expandafter| endgroup \the \toks@ #1%|
1179
1180 }% \tabu@endofcollect
```

11.11 One trial after the other (\tabu@strategy)

Switching between the strategies

\tabu@strategy

This macro does some specific setup depending on the strategy (1, 2 or 3), and orders to finish when all measurements are done.

This consists in a switch (\ifcase) which is done before the trials by \tabu@strategy, and after the trials by \tabu@endtrial.

```
1181 \def\tabu@strategy {\relax % stops \count@ assignment !
                                 % case 0 = print with vertical adjustment (outer is finished)
1182
        \ifcase\count@
1183
            \expandafter \tabu@endoftrials
1184
        \or
                                 % case 1 = exit in vertical measure (outer in mode 3)
1185
            \expandafter\xdef\csname tabu@\the\tabu@nested.T\endcsname{\the\tabu@target}%
            \expandafter\xdef\csname tabu@\the\tabu@nested.X\endcsname{\the\tabucolX}%
1186
1187
            \expandafter \tabu@endoftrials
1188
                                 % case 2 = exit with a rule replacing the table (outer in mode 4)
        \or
```

```
1189
             \expandafter \tabu@quickend
1190
                                  % case 3 = outer is in mode 3 because of no X
        \or
1191
            \begingroup
1192
                 \tabu@evr{\tabu@verticalinit}\tabu@celllalign@def{\tabu@verticalmeasure}%
1193
                 \def\tabu@cellralign{\tabu@verticalspacing}%
1194
                 \expandafter \tabu@measuring
1195
        \else
                                  % case 4 = horizontal measure
1196
            \begingroup
                 \global\let\tabu@elapsedtime \tabu@message@etime
1197
                 \label{longdef} $$  \omega= 1 + 1 + 2 + 3 {\mathbf multispan} {\#1} }
1198
                 \let\tabu@startpboxORI \@startpbox
1199
1200
                 \iftabu@spread
                         \def\tabu@naturalXmax {\z@}%
1201
1202
                         \let\tabu@naturalXmin \tabu@naturalXmax
1203
                         \tabu@evr{\global\tabu@naturalX \z@}%
                         \let\@startpbox \tabu@startpboxmeasure
1204
                 \else\iftabu@negcoef
1205
1206
                         \let\@startpbox \tabu@startpboxmeasure
                         \let\@startpbox \tabu@startpboxquick
1207
                 \else
1208
                 \fi\fi
                 \expandafter \tabu@measuring
1209
1210
        \fi
1211 }% \tabu@strategy
```

\tabu@measuring

Expands \tabu@trial with the whole content of the environment stored in \toks@ by \tabu@collectbody.

At the end of the trial, \count@ will be reassigned to the value it had before the trial. Then \tabu@endtrial will choose the algorithm depending on the strategy number, and set the new strategy number (into \count@ again) for the next step.

\tabu@trial

This is the starting point of trials: \halign is expanded here.

\tabu@longtrial

This is the long version of \tabu@trial for longtabu. Almost the same apart for the math group and the end (a longtable environment does not finish with \endarray).

```
1212 \def\tabu@measuring{\expandafter \tabu@trial \expandafter
1213
                                                      \count@ \the\count@ \tabu@endtrial
1214 }% \tabu@measuring
1215 \def\tabu@trial{\iftabu@long \tabu@longtrial \else \tabu@shorttrial \fi}
1216 \def\tabu@shorttrial {\setbox\tabu@box \hbox\bgroup \tabu@seteverycr
1217
        \ifx \tabu@savecounters\relax \else
1218
                    \let\tabu@savecounters \relax \tabu@clckpt \fi
1219
        $\iftabuscantokens \tabu@rescan \else \expandafter\@secondoftwo \fi
           \expandafter{\expandafter \tabuthepreamble
1220
1221
                              \the\tabu@thebody
1222
                              \csname tabu@adl@endtrial\endcsname
1223
                              \endarray}$\egroup
                                                              % got \tabu@box
1224 }% \tabu@shorttrial
1225 \def\tabu@longtrial {\setbox\tabu@box \hbox\bgroup \tabu@seteverycr
        \ifx \tabu@savecounters\relax \else
1226
1227
                    \let\tabu@savecounters \relax \tabu@clckpt \fi
1228
        \iftabuscantokens \tabu@rescan \else \expandafter\@secondoftwo \fi
1229
           \expandafter{\expandafter \tabuthepreamble
1230
                              \the\tabu@thebody
1231
                              \tabuendlongtrial}\egroup
                                                              % got \tabu@box
1232 }% \tabu@longtrial
1233 \det \text{tabuendlongtrial} % no @ allowed for \scantokens
1234
        \LT@echunk \global\setbox\@ne \hbox{\unhbox\@ne}\kern\wd\@ne
1235
                    \LT@get@widths
1236 }% \tabuendlongtrial
```

```
1237 \end{add} \end{trial} {\it $\arydshln in nested trials - problem for global column counters!} \\ 1238 \end{add} \end{addd} \arydshln in nested trials - problem for global column counters!} \\ 1238 \end{adddense} \
```

\tabu@seteverycr

\ialign resets \everycr to an empty token. This macro sets \everycr for the tabu environment : a bridge around \ialign is built: \everycr redefines itself \afterassignment!

```
1239 \def\tabu@seteverycr {\tabu@reset

1240 \everycr \expandafter{\the\everycr \tabu@everycr}%

1241 \let\everycr \tabu@noeverycr % <for ialign>

1242 }% \tabu@seteverycr

1243 \def\tabu@noeverycr{{\aftergroup\tabu@restoreeverycr \afterassignment}\toks@}

1244 \def\tabu@restoreeverycr {\let\everycr \tabu@everycr}

1245 \def\tabu@everycr {\iftabu@everyrow \noalign{\tabu@everyrow}\fi}
```

\tabu@endoftrials

When the algorithm said the tabular was ready to be printed, **\tabu@endoftrials** closes the trials group and prints the tabular...

The required values (column widths, struts etc.) are *injected* into the group by the mean of the buffer \tabu@bufferX (locally defined).

\tabu@closetrialsgroup This closes the group in which all the trials are done.

```
1246 \def\tabu@endoftrials {%
1247
        \iftabuscantokens
                             \expandafter\@firstoftwo
1248
        \else
                             \expandafter\@secondoftwo
1249
        \fi
            {\expandafter \tabu@closetrialsgroup \expandafter
1250
             \tabu@rescan \expandafter{%
1251
1252
                         \expandafter\tabuthepreamble
1253
                             \the\expandafter\tabu@thebody
1254
                                          \iftabu@long \else \endarray \fi}}
1255
            {\expandafter\tabu@closetrialsgroup \expandafter
1256
                         \tabuthepreamble
1257
                             \the\tabu@thebody}%
1258
                                              \tabu@endenvir
                                                                   % Finish !
1259 }% \tabu@endoftrials
1260 \def\tabu@closetrialsgroup {%
1261
        \toks@\expandafter{\tabu@endenvir}%
1262
        \edef\tabu@bufferX{\endgroup
1263
            \t.abucolX
                             \the\tabucolX
1264
            \tabu@target
                             \the\tabu@target
1265
            \t.abu@cnt
                             \the\tabu@cnt
1266
            \def\noexpand\tabu@endenvir{\the\toks@}%
            %Quid de \@halignto = \tabu@halignto ??
1267
1268
        }% \tabu@bufferX
        \tabu@bufferX
1269
1270
        \ifcase\tabu@nested % print out (outer in mode 0)
            \global\tabu@cnt \tabu@cnt
1271
1272
            \tabu@evr{\tabu@verticaldynamicadjustment}%
1273
            \tabu@celllalign@def{\everypar{}}\let\tabu@cellralign \@empty
1274
            \let\@finalstrut \tabu@finalstrut
1275
        \else
                             % vertical measure of nested tabu
1276
            \tabu@evr{\tabu@verticalinit}%
1277
            \tabu@celllalign@def{\tabu@verticalmeasure}%
1278
            \def\tabu@cellralign{\tabu@verticalspacing}%
1279
        \tabu@clckpt \let\@halignto \tabu@halignto
1280
1281
        \let\@halignto \@empty
1282
        \tabu@seteverycr
1283
        \ifdim \tabustrutrule>\z@ \ifnum\tabu@nested=\z@
1284
            \setbox\@arstrutbox \box\voidb@x % force \@arstrutbox to be rebuilt (visible struts)
```

```
1285
        \fi\fi
1286 }% \tabu@closetrialsgroup
```

\tabu@quickend

Quick exit after having measuring the natural width of a nested tabu.

```
1287 \def\tabu@quickend {\expandafter \endgroup \expandafter
                            \tabu@target \the\tabu@target \tabu@quickrule
1288
1289
                             \let\endarray \relax \tabu@endenvir
1290 }% \tabu@quickend
```

\tabu@endtrial

Depending on the strategy that was just applied, \tabu@endtrial chooses the algorithm and determines the number of the strategy for the next step.

```
1291 \def\tabu@endtrial {\relax
                                                                                                                   % stops \count@ assignment !
1292
                        \ifcase \count@ \tabu@err
                                                                                                                  % case 0 = impossible here
1293
                                                                            \tabu@err
                                                                                                                  % case 1 = impossible here
                         \or
1294
                         \or
                                                                            \tabu@err
                                                                                                                  % case 2 = impossible here
1295
                        \or
                                                                                                                   % case 3 = outer goes into mode 0
1296
                                      \def\tabu@bufferX{\endgroup}\count@ \z@
                                                                                                                   % case 4 = outer goes into mode 3
1297
1298
                                      \iftabu@spread \tabu@spreadarith % inner into mode 1 (outer in mode 3)
1299
                                      \else
                                                                                        \tabu@arith
                                                                                                                                                                                                   or 2 (outer in mode 4)
                                      \fi
1300
1301
                                      \count@=%
1302
                                                  \ifcase\tabu@nested
                                                                                                                                \thr@@ % outer goes into mode 3
                                                  \else\iftabu@measuring
                                                                                                                                                          % outer is in mode 4
1303
                                                                                                                               \tw@
1304
                                                  \else
                                                                                                                                \@ne
                                                                                                                                                          % outer is in mode 3
                                                  \fi\fi
1305
                                      \edef\tabu@bufferX{\endgroup
1306
1307
                                                                                                   \tabucolX
                                                                                                                                                          \the\tabucolX
                                                                                                                                                          \the\tabu@target}%
1308
                                                                                                    \tabu@target
1309
1310
                         \expandafter \tabu@bufferX \expandafter
1311
                                                                                                                                \count@ \the\count@ \tabu@strategy
1312 }% \tabu@endtrial
1313 \ensuremath{\mbox{\mbox{$1$}}} error! \ensuremath{\mbox{\mbox{$1$}}} error! \ensuremath{\mbox{$1$}} error! \ensuremat
```

11.12 The algorithms: Measuring the tabu box

At the end of each trial, we call \tabu@arith (or \tabu@spreadarith) to computes the widths and update the values.

At the exit, \iftabu@measuring is set to \iftrue: a further trial is necessary, or \iffalse: the target width is reached.

The arithmetic of X columns: the tabu to case

\tabu@arithnegcoef This is a loop against the width coefficients. There is no \@for or \@whiles because \tabu@Xcoefs stores the series in the form: \tabu@ {coef1} \tabu@ {coef2} \tabu@ {coef3}.

Thus, just \let \tabu@ to be \tabu@arith@negcoef and expand \tabu@Xcoefs!

The aim of the game is to *neutralize* some X columns: when their natural width are less than coef×\tabucolX.

```
1314 \def\tabu@arithnegcoef {%
        \@tempdima \z@ \dimen@ \z@ \let\tabu@ \tabu@arith@negcoef \tabu@Xcoefs
1316 }% \tabu@arithnegcoef
1317 \def\tabu@arith@negcoef #1#2{%
1318
        \left| \frac{42}{p_0} \right|
                             \advance\dimen@
                                                   #2\p@
                                                                % saturated by definition
1319
                              \advance\@tempdima #2\tabucolX
1320
        \else
1321
            \ifdim -#2\tabucolX <\tabu@wd{#1}% c_i X < natural width <= \tabu@target-> saturated
                                                   -#2\p@
1322
                              \advance\dimen@
```

```
1323
                                                           \advance\@tempdima -#2\tabucolX
                  1324
                                    \else
                  1325
                                                           \advance\@tempdima \tabu@wd{#1}% natural width <= c_i X => neutralised
                                                           \ifdim \tabu@wd{#1}<\tabu@target \else % neutralised
                  1326
                  1327
                                                                                       -#2\p@ % saturated (natural width = tabu@target)
                                                           \advance\dimen@
                  1328
                                                           \fi
                  1329
                                    \fi
                  1330
                              \fi
                  1331 }% \tabu@arith@negcoef
\tabu@arith General algorithms for tabu to with X columns.
                  1332 \def\tabu@givespace #1#2{% here \tabu@DELTA < \z@
                              \ifdim \@tempdima=\z@
                  1333
                  1334
                                    \tabu@wddef{#1}{\the\dimexpr -\tabu@DELTA*\p@/\tabu@Xsum}%
                  1335
                              \else
                                    \tabu@wddef{#1}{\the\dimexpr \tabu@hsize{#1}{#2}
                  1336
                  1337
                                                     *(\p@ -\tabu@DELTA*\p@/\@tempdima)/\p@\relax}%
                              \fi
                  1338
                  1339 }% \tabu@givespace
                  1340 \ensuremath{\def \advance \ensuremath{\decnt \ensuremath{\decnt}\decnt}} \ensuremath{\decnt \ensuremath{\decnt}\decnt} \ensuremath{\decnt \ensuremath{\decnt}\decnt} \ensuremath{\decnt}\decnt \ensuremath{\decnt}\decnt} \ensuremath{\decnt \ensuremath{\decnt}\decnt} \ensuremath{\decnt}\decnt \ensuremath{\decnt}\decnt} \ensuremath{\decnt}\decnt
                  1341
                              \ifnum \tabu@cnt=\@ne \tabu@message{\tabu@titles}\fi
                  1342
                              \tabu@arithneqcoef
                  1343
                              \@tempdimb \dimexpr \wd\tabu@box -\@tempdima \relax % <incompressible material>
                              \tabu@DELTA = \dimexpr \wd\tabu@box - \tabu@target \relax
                  1344
                  1345
                              \tabu@message{\tabu@message@arith}%
                              \ifdim \tabu@DELTA <\tabu@hfuzz
                  1346
                  1347
                                    \ifdim \tabu@DELTA<\z@
                                                                                  % wd (tabu) < \tabu@target ?
                  1348
                                         \let\tabu@\tabu@givespace \tabu@Xcoefs
                  1349
                                          \advance\@tempdima \@tempdimb \advance\@tempdima -\tabu@DELTA % for message
                  1350
                                    \else
                                               % already converged: nothing to do but nearly impossible...
                                    \fi
                  1351
                                    \tabucolX \maxdimen
                  1352
                  1353
                                    \tabu@measuringfalse
                  1354
                              \else
                                                                                   % need for narrower X columns
                  1355
                                    \tabucolX =\dimexpr (\@tempdima -\tabu@DELTA) *\p@/\tabu@Xsum \relax
                  1356
                                    \tabu@measuringtrue
                  1357
                                    \@whilesw \iftabu@measuring\fi {%
                                         \advance\tabu@cnt \@ne
                  1358
                  1359
                                          \tabu@arithneqcoef
                                         \tabu@DELTA =\dimexpr \@tempdima+\@tempdimb -\tabu@target \relax % always < 0 here
                  1360
                  1361
                                         \tabu@message{\tabu@header
                                               \tabu@msgalign \tabucolX { }{ }{ }{ }\@@
                  1362
                                               1363
                                               \tabu@msgalign \tabu@target { }{ }{ }{ }\@@
                  1364
                  1365
                                               \tabu@msgalign@PT \dimen@ { }{}{}{}{}{}\@@
                  1366
                                               \ifdim -\tabu@DELTA<\tabu@hfuzz \tabu@spaces target ok\else
                  1367
                                               \t \ \tabu@msgalign \dimexpr -\tabu@DELTA *\p@/\dimen@ {}{}{}{}{}\@@
                  1368
                                               \fi}%
                                         \ifdim -\tabu@DELTA<\tabu@hfuzz
                  1369
                  1370
                                               \advance\@tempdima \@tempdimb % for message
                  1371
                                               \tabu@measuringfalse
                  1372
                                         \else
                  1373
                                               \advance\tabucolX \dimexpr -\tabu@DELTA *\p@/\dimen@ \relax
                  1374
                                         \fi
                                   } 응
                  1375
                  1376
                              \fi
                  1377
                              \tabu@message{\tabu@message@reached}%
                  1378
                              \edef\tabu@bufferX{\endgroup \tabu@cnt
                                                                                           \the\tabu@cnt
```

The arithmetic of X columns for tabu spread

\tabu@spreadarith

Algorithm for tabu spread with X columns: the aim of the game is to compute the target (relative to the natural width of the tabular) and go to \tabu@arith afterwards.

```
1382 \def\tabu@spreadarith {%
1383
        \dimen@ \z@ \@tempdima \tabu@naturalXmax \let\tabu@ \tabu@spread@arith \tabu@Xcoefs
1384
        \edef\tabu@naturalXmin {\the\dimexpr\tabu@naturalXmin*\dimen@/\p@}%
1385
        \@tempdimc =\dimexpr \wd\tabu@box -\tabu@naturalXmax+\tabu@naturalXmin \relax
1386
        \iftabu@measuring
            \tabu@target =\dimexpr \@tempdimc+\tabu@spreadtarget \relax
1387
1388
            \edef\tabu@bufferX{\endqroup \tabucolX \tabucolX \tabu@target\the\tabu@target}}
1389
        \else
1390
            \tabu@message{\tabu@message@spreadarith}%
1391
            \ifdim \dimexpr \@tempdimc+\tabu@spreadtarget >\tabu@target
                \tabu@message{(tabu) spread
1392
1393
                    \ifdim \@tempdimc>\tabu@target useless here: default target used%
1394
                    \else too large: reduced to fit default target\fi.}%
1395
            \else
                \tabu@target =\dimexpr \@tempdimc+\tabu@spreadtarget \relax
1396
1397
                \tabu@message{(tabu) spread: New target set to \the\tabu@target^^J}%
1398
            \fi
            \begingroup \let\tabu@wddef \@gobbletwo
1399
1400
                \@tempdimb \@tempdima
                \tabucolX@init
1401
1402
                \tabu@arithnegcoef
1403
                \wd\tabu@box =\dimexpr \wd\tabu@box +\@tempdima-\@tempdimb \relax
            \expandafter\endgroup \expandafter\tabucolX \the\tabucolX
1404
1405
            \tabu@arith
1406
        \fi
1407 }% \tabu@spreadarith
1408 \def\tabu@spread@arith #1#2{%
        \ifdim #2\p@>\z@ \advance\dimen@ #2\p@
1409
1410
        \else
                          \advance\@tempdima \tabu@wd{#1}\relax
1411
        \fi
1412 }% \tabu@spread@arith
```

Reporting in the .log file (debugshow option)

\tabu@message@defaulttarget

```
1413 \def\tabu@message@defaulttarget {%
1414
       \ifnum\tabu@nested=\z@^^J(tabu) Default target:
1415
       \ifx\tabudefaulttarget\linewidth
                                       \string\linewidth
           \ifdim \tabu@thetarget=\linewidth \else
1416
1417
              -\the\dimexpr\linewidth-\tabu@thetarget\fi =
1418
       \else\ifx\tabudefaulttarget\linegoal\string\linegoal=
1419
       \fi\fi
1420
       \else (tabu) Default target (nested): \fi
1421
       \the\tabu@target \on@line
       1422
1423 \def\tabu@message@target {^^J(tabu) Target specified:
      \the\tabu@target \on@line, page \the\c@page}
```

 $\begin{array}{c} \mathbf{1} \\ \mathbf{$

1425 \def\tabu@message@arith {\tabu@header

```
1426 \tabu@msgalign \tabucolX { }{ }{ }{ }\@@

1427 \tabu@msgalign \wd\tabu@box { }{ }{ }\@@

1428 \tabu@msgalign \tabu@target { }{ }\@@

1429 \tabu@msgalign@PT \dimen@ { }{}{}{}\@@

1430 \ifdim \tabu@DELTA<\tabu@hfuzz giving space\else

1431 \tabu@msgalign \dimexpr (\@tempdima-\tabu@DELTA) *\p@/\tabu@Xsum -\tabucolX {}{}{}\%\@@

1432 \fi

1433 }% \tabu@message@arith
```

\tabu@message@spreadarith

\tabu@message@negcoef

```
1442 \def\tabu@message@negcoef #1#2{
1443
        \tabu@spaces\tabu@spaces\space * #1. X[\rem@pt#2]:
1444
        \space width = \tabu@wd {#1}
1445
            \expandafter\string\csname tabu@\the\tabu@nested.W\number#1\endcsname
1446
        \ifdim -\tabu@pt#2\tabucolX<\tabu@target
        < \number-\rem@pt#2 X
1447
1448
        = \the\dimexpr -\tabu@pt#2\tabucolX \relax
1449
1450
        <= \the\tabu@target\space < \number-\rem@pt#2 X\fi}</pre>
```

\tabu@message@reached

\tabu@message@etime

```
1454 \def\tabu@message@etime{\edef\tabu@stoptime{\the\pdfelapsedtime}%

1455 \tabu@message{(tabu)\tabu@spaces Time elapsed during measure:

1456 \the\numexpr(\tabu@stoptime-\tabu@starttime-32767)/65536\relax sec

1457 \the\numexpr\numexpr(\tabu@stoptime-\tabu@starttime)

1458 -\numexpr(\tabu@stoptime-\tabu@starttime-32767)/65536\relax*65536\relax

1459 *1000/65536\relax ms \tabu@spaces(\the\tabu@cnt\space

1460 cycle\ifnum\tabu@cnt>\@ne s\fi)^^J}}
```

\tabu@message@verticalsp

```
1461 \def\tabu@message@verticalsp {%
1462
      \ifdim \@tempdima>\tabu@ht
          \ifdim \@tempdimb>\tabu@dp
1463
1464
          \expandafter\expandafter\expandafter\string\tabu@ht =
             \tabu@msgalign \@tempdima { }{ }{ }{ }\@@
1465
1466
          \expandafter\expandafter\expandafter\string\tabu@dp =
1467
             1468
          \expandafter\expandafter\expandafter\string\tabu@ht =
1469
1470
             1471
          \fi
1472
      \else\ifdim \@tempdimb>\tabu@dp
1473
          \tabu@spaces\tabu@spaces\tabu@spaces
```

\tabu@message@save

```
1478 \edef\tabu@spaces { \@spaces }
1479 \ensuremath{\verb| def \times | \text{lexpandafter tabu@pt the}|}
1480 {\@makeother\P\\@makeother\T\lowercase{\qdef\tabu@pt #1PT{#1}}}
1481 \def\tabu@msqalign{\expandafter\tabu@msq@align\the\dimexpr}
1482 \def\tabu@msgalign@PT{\expandafter\tabu@msg@align\romannumeral-'\0\tabu@strippt}
1483 \leq \#1{\%}
1484
       \def\tabu@msg@align##1.##2##3##4##5##6##7##8##9\@@{%
       \ifnum##1<10 #1 #1\else
1485
1486
       \ifnum##1<100 #1 \else
       1487
1488
       ##1.##2##3##4##5##6##7##8#1}%
1489
       \def\tabu@header{(tabu) \ifnum\tabu@cnt<10 #1\fi\the\tabu@cnt) }%</pre>
1490
       \def\tabu@titles{\ifnum \tabu@nested=\z@
         (tabu) Try#1 #1 tabu X #1 #1 #1tabu Width #1 #1 Target
1491
1492
                     #1 #1 #1 Coefs #1 #1 #1 Update^^J\fi}%
1493
       \def\tabu@spreadheader{%
1494
         (tabu) Try#1 #1 Spread #1 #1 tabu Width #1 #1 Nat. X #1 #1 #1 #1Nat. Min.
1495
                                                         #1 New Target^^J%
1496
         (tabu) sprd}
1497
       \def\tabu@message@save {\begingroup
1498
           \def\x ####1{\tabu@msg@align ####1{ }{ }{ }{ }{ }{\}@@}
1499
           \def\z ####1{\expandafter\x\expandafter{\romannumeral-'\0\tabu@strippt
1500
                                                        \dimexpr####1\p@{ }{ }}}%
1501
           \let\color \relax \def\tabu@rulesstyle ####1###2{\detokenize{####1}}%
           \let\CT@arc@ \relax \let\@preamble \@gobble
1502
1503
           \let\tabu@savedpream \@firstofone
1504
           \let\tabu@savedparams \@firstofone
                                          {(tabu) target #1 #1 #1 #1 #1 = x{\#\#\#1}^{J}
1505
           \def\tabu@target ####1\relax
1506
           \def\tabucolX ####1\relax
                                          {(tabu) X columns width#1 = x{\#\#\#1}^{J}
           \def\tabu@nbcols ####1\relax
                                          {(tabu) Number of columns: z{\#\#\#1}^{J}}
1507
           \def\tabu@aligndefault
1508
                                     ####1{(tabu) Default alignment: #1 #1 ####1^^J}%
1509
           \def\col@sep ####1\relax
                                          {(tabu) column sep #1 #1 #1 = x{\#\#\#1}^{5}
1510
           \def \arrayrulewidth ####1\relax{(tabu) arrayrulewidth #1 = \x{####1}}%
           \def\doublerulesep ####1\relax { doublerulesep = <math>x{####1}^3}
1511
1512
           \def\extratabsurround####1\relax{(tabu) extratabsurround = \x{####1}^^J}%
           1513
1514
           \def\ensuremath{\def}\ {extrarowdepth = \x{\#\#\#1}^{\n}}
           \def\abovetabulinesep####1\relax{(tabu) abovetabulinesep=\x{####1} }%
1515
1516
           \def\belowtabulinesep####1\relax{ belowtabulinesep=\x{####1}^^J}%
1517
                                     ####1{(tabu) arraystretch #1 #1 = \z{####1}^^J}%
           \def\arraystretch
1518
           \def\minrowclearance ####1\relax{(tabu) minrowclearance #1 = \x{####1}^^J}%
1519
                                     ####1{(tabu) taburulecolor #1 #1 = ####1^^J}%
           \def\tabu@arc@L
                                     ####1{(tabu) tabudoublerulecolor= ####1^^J}%
1520
           \def\tabu@drsc@L
1521
           \def\tabu@evr@L
                                    ####1{(tabu) everyrow #1 #1 #1 #1 = \detokenize{####1}^^J}%
1522
           \def\tabu@ls@L
                                     ####1{(tabu) line style = \detokenize{####1}^^J}%
1523
           \label{localized} $$ \left( \frac{\#\#\#1}{2\pi} \right) = \frac{\#\#\#1}{2\pi} . $$
           1524
           \let\edef \@gobbletwo \let\def \@empty \let\let \@gobbletwo
1525
1526
           \tabu@message{%
1527
            (tabu) \string\savetabu{\tabu@temp}: \on@line^^J%
1528
            \tabu@usetabu \@nil^^J}%
1529
           \endgroup}
```

11.13 Measuring the natural width of columns (varwidth code from D. Arseneau)

\tabu@startpboxmeasure The important job is done at the end: by \tabu@endpboxmeasure.

When "tabu spread" is used with X columns, the first trial must measure the natural width of the columns. When X columns have negativ coefficient, the natural is computed after the target has been reached, with the absolute coefficients.

Nested trials may occur (tabu spread inside a X column with negativ coefficient for example).

For the furthur trials, the standard scheme for X column is used: the natural width is measured only once.

pdfT_EX font expansion is disabled inside the varwidth environment (we set \pdfadjustspacing to 0).

```
1531 \def\tabu@startpboxmeasure #1{\bgroup
                                         % entering \vtop
       \edef\tabu@temp{\expandafter\@secondoftwo \ifx\tabu@hsize #1\else\relax\fi}%
1532
       \ifodd 1\ifx \tabu@temp\@empty 0 \else
1533
                                                  % starts with \tabu@hsize ?
               \iftabu@spread
                                        \else
                                                   % if spread -> measure
1534
               \ifdim \tabu@temp\p@>\z@ 0 \fi\fi\fi\ if coef>0 -> do not measure
1535
1536
           \let\@startpbox \tabu@startpboxORI
                                                   % restore immediately (nesting)
           \tabu@measuringtrue
                                                   % for the quick option...
1537
           \tabu@Xcol =\expandafter\@firstoftwo\ifx\tabu@hsize #1\fi
1538
           1539
                                           \tabu@target=\tabu@temp\tabucolX \fi\fi
1540
1541
           \setbox\tabu@box \hbox \bgroup
1542
               \begin{varwidth}\tabu@target
1543
                   \let\FV@ListProcessLine \tabu@FV@ListProcessLine % \hbox to natural width...
                   \narrowragged \arraybackslash \parfillskip \@flushglue
1544
                   \ifdefined\pdfadjustspacing \pdfadjustspacing\z@ \fi
1545
1546
                   \bgroup \aftergroup\tabu@endpboxmeasure
                   \ifdefined \cellspacetoplimit \tabu@cellspacepatch \fi
1547
1548
       \else \expandafter\@gobble
1549
                               \tabu@startpboxquick{#1}% \@gobble \bgroup
       \fi
1550
1551 }% \tabu@startpboxmeasure
1552 \def\tabu@cellspacepatch{\def\bcolumn##1\@nil{}\let\ecolumn\@empty
1553
                                             \bgroup\color@begingroup}
```

\tabu@endpboxmeasure The cell has been built inside a box: we have to get its dimensions, and update \tabu@naturalX, \tabu@naturalXmin and \tabu@naturalXmax accordingly (for tabu spread), and even store (globally) each column width: the column width is the maximum width of the cells it contains.

```
1554 \def\tabu@endpboxmeasure {%
        \@finalstrut \@arstrutbox
1555
1556
                         \end{varwidth}\egroup
                                                    % <got my \tabu@box>
1557
        \left( \frac{1}{2} \right) = \frac{1}{2}
                                      % neg coef
            \ifdim \tabu@wd\tabu@Xcol <\wd\tabu@box
1558
1559
                \tabu@wddef\tabu@Xcol {\the\wd\tabu@box}%
1560
                \tabu@debug{\tabu@message@endpboxmeasure}%
            \fi
1561
1562
        \else
                                      % spread coef>0
1563
            \global\advance \tabu@naturalX \wd\tabu@box
            \@tempdima =\dimexpr \wd\tabu@box *\p@/\dimexpr \tabu@temp\p@\relax \relax
1564
1565
            \ifdim \tabu@naturalXmax <\tabu@naturalX
                \xdef\tabu@naturalXmax {\the\tabu@naturalX}\fi
1566
1567
            \ifdim \tabu@naturalXmin <\@tempdima
1568
                \xdef\tabu@naturalXmin {\the\@tempdima}\fi
1569
        \fi
1570
       \box\tabu@box \egroup % end of \vtop (measure) restore \tabu@target
```

```
1571 }% \tabu@endpboxmeasure
1572 \def\tabu@wddef #1{\expandafter\xdef
1573
                       \csname tabu@\the\tabu@nested.W\number#1\endcsname}
                    #1{\csname tabu@\the\tabu@nested.W\number#1\endcsname}
1574 \def\tabu@wd
1575 \def\tabu@message@endpboxmeasure{\tabu@spaces\tabu@spaces<-> % <-> save natural wd
1576
        \the\tabu@Xcol. X[\tabu@temp]:
1577
       target = \the\tabucolX \space
1578
        \expandafter\expandafter\expandafter\string\tabu@wd\tabu@Xcol
1579
        =\tabu@wd\tabu@Xcol
1580 \ \tabu@message@endpboxmeasure
```

\tabu@startpboxquick Contents of paragraph columns are not built during trials in strategy number 4.

```
1581 \def\tabu@startpboxquick {\bgroup
       \let\@startpbox \tabu@startpboxORI % restore immediately
1582
       \let\tabu \tabu@quick
1583
                                          % \begin is expanded before...
1584
       \expandafter\@gobble \@startpbox
                                          % gobbles \bgroup
1585 }% \tabu@startpboxquick
1586 \def\tabu@quick {\begingroup \iffalse{\fi \ifnum0='}\fi
       \toks@{}\def\tabu@stack{b}\tabu@collectbody \tabu@endquick
1587
1588 }% \tabu@quick
1589 \def\tabu@endquick {%
1590
       \ifodd 1\ifx\tabu@end@envir\tabu@endtabu \else
1591
               1592
               \endgroup
1593
       \else
               \let\endtabu \relax
               \tabu@end@envir
1594
       \fi
1595
1596 }% \tabu@quick
1597 \def\tabu@endtabu
                       {\end{tabu}}
1598 \def\tabu@endtabus
                      {\end{tabu*}}
```

11.14 Measuring the height and depths of rows

\tabu@verticalmeasure Starting point for vertical measure of every cell. Only the maxima/minima are stored, for $\tau_{\aleph}b \subset \text{must}$ know the height/depth of every row.

```
1599 \def\tabu@verticalmeasure{\everypar{}%
1600
        \ifnum \currentgrouptype>12
                                              % 14=semi-simple, 15=math shift group
1601
            \setbox\tabu@box =\hbox\bgroup
1602
                \let\tabu@verticalspacing \tabu@verticalsp@lcr
1603
                \d@llarbegin
                                              % after \hbox ...
1604
        \else
1605
            \edef\tabu@temp{\ifnum\currentgrouptype=5\vtop
1606
                             \else\ifnum\currentgrouptype=12\vcenter
                             \else\vbox\fi\fi}%
1607
1608
            \setbox\tabu@box \hbox\bgroup$\tabu@temp \bgroup
1609
                \let\tabu@verticalspacing \tabu@verticalsp@pmb
1610
        \fi
1611 }% \tabu@verticalmeasure
```

\tabu@verticalsp@lcr Vertical spacing adjustment for standard 1, c, r columns.

```
1612 \def\tabu@verticalsp@lcr{%
1613
        \d@llarend \egroup
                                  % <got my \tabu@box>
1614
        \@tempdima \dimexpr \ht\tabu@box+\abovetabulinesep
1615
        \@tempdimb \dimexpr \dp\tabu@box+\belowtabulinesep \relax
1616
            \ifdim\tabustrutrule>\z@ \tabu@debug{\tabu@message@verticalsp}\fi
1617
        \ifdim \tabu@ht<\@tempdima
                                       \tabu@htdef{\the\@tempdima}\fi
1618
        \ifdim \tabu@dp<\@tempdimb
                                       \tabu@dpdef{\the\@tempdimb}\fi
1619
        \noindent\vrule height\@tempdima depth\@tempdimb
```

```
1620 }% \tabu@verticalsp@lcr
```

\tabu@verticalsp@pmb Vertical spacing adjustment with struts for p, m, or b columns.

```
1621 \ensuremath{\mbox{\sc loss}} \ensurema
                                  \par \expandafter\egroup
1622
1623
                                                                    \expandafter$\expandafter
1624
                                                                                                       \egroup \expandafter
                                                                                                                                         \@tempdimc \the\prevdepth
1625
1626
                                  \@tempdima \dimexpr \ht\tabu@box+\abovetabulinesep
1627
                                  \@tempdimb \dimexpr \dp\tabu@box+\belowtabulinesep \relax
1628
                                                   \ifdim\tabustrutrule>\z@\tabu@debug{\tabu@message@verticalsp}\fi
1629
                                  \ifdim \tabu@ht<\@tempdima
                                                                                                                                                                    \tabu@htdef{\the\@tempdima}\fi
                                 \ifdim \tabu@dp<\@tempdimb
                                                                                                                                                                    \tabu@dpdef{\the\@tempdimb}\fi
1630
1631
                                 \let\@finalstrut \@gobble
1632
                                  \hrule height\@tempdima depth\@tempdimb width\hsize
1633 %%
                                          \box\tabu@box
1634 }% \tabu@verticalsp@pmb
```

\tabu@verticalinit Initialisation of \tabu@ht and \tabu@dp. Done at \everyrow.

```
1635 \def\tabu@verticalinit{%
1636
                                 \ifnum \c@taburow=\z@ \tabu@rearstrut \fi
                                                                                                                                                                                                                                                      % after \tabu@reset !
1637
                                  \advance\c@taburow \@ne
1638
                                  \tabu@htdef{\the\ht\@arstrutbox}\tabu@dpdef{\the\dp\@arstrutbox}%
1639
                                  \advance\c@t.aburow \m@ne
1640 }% \tabu@verticalinit
1641 \def\tabu@htdef {\expandafter\xdef \csname tabu@\the\tabu@nested.H\the\c@taburow\endcsname}
1642 \leq f 
                                                                                                                                                                      {\csname tabu@\the\tabu@nested.H\the\c@taburow\endcsname}
1643 \end{figure} $$1643 \end{figure} \end{figure} $$1643 \end{f
1644 \def\tabu@dp
                                                                                                                                                                      {\csname tabu@\the\tabu@nested.D\the\c@taburow\endcsname}
```

\tabu@verticaldynamicadjustment This updates the \@arstrutbox at \everyrow (ie.\everycr) in order to adjust the vertical spacing of cells.

```
1645 \def\tabu@verticaldynamicadjustment {%
1646 \advance\c@taburow \@ne
1647 \extrarowheight \dimexpr\tabu@ht - \ht\strutbox
1648 \extrarowdepth \dimexpr\tabu@dp - \dp\strutbox
1649 \let\arraystretch \@empty
1650 \advance\c@taburow \m@ne
1651 }% \tabu@verticaldynamicadjustment
```

11.15 \tabuphantomline

\tabuphantomline This macro inserts a phantom line in front of a tabu. This is necessary when you use \usetabu with tabu X column, with a single line containing \multicolumn...

```
1652 \def \tabuphantomline{\crcr \noalign{%}
1653
        {\globaldefs \@ne
1654
            \setbox\@arstrutbox
                                      \box\voidb@x
1655
            \let\tabu@@celllalign
                                      \tabu@celllalign
1656
            \let\tabu@@cellralign
                                      \tabu@cellralign
            \let\tabu@@cellleft
                                      \tabu@cellleft
1657
1658
            \let\tabu@@cellright
                                      \tabu@cellright
1659
            \let\tabu@@thevline
                                      \tabu@thevline
1660
            \let\tabu@celllalign
                                      \@empty
1661
            \let\tabu@cellralign
                                      \@empty
            \let\tabu@cellright
1662
                                      \@empty
1663
            \let\tabu@cellleft
                                      \@empty
1664
            \let\tabu@thevline
                                      \relax}%
1665
        \edef\tabu@temp{\tabu@multispan \tabu@nbcols{\noindent &}}%
```

```
\toks@\expandafter{\tabu@temp \noindent\tabu@everyrowfalse \cr
1666
1667
            \noalign{\tabu@rearstrut
1668
                {\globaldefs\@ne
                    \let\tabu@celllalign \tabu@@celllalign
1669
                    \let\tabu@cellralign \tabu@@cellralign
1670
1671
                    \let\tabu@cellleft \tabu@@cellleft
                    \let\tabu@cellright \tabu@@cellright
1672
1673
                    \let\tabu@thevline \tabu@@thevline}}}%
1674
        \expandafter}\the\toks@
1675 }% \tabuphantomline
```

11.16 Horizontal lines inside tabu: \tabucline, \firsthline and \lasthline

Horizontal lines: multiple \firsthline / \lasthline

\tabu@firstline
\tabu@lastline
\tabu@firsthline
\tabu@lasthline

\firsthline and \lasthline are \let to \tabu@firsthline and \tabu@lasthline inside the tabu environment.

This allows to duplicate horizontal lines, while keeping the alignement:

\firsthline \firsthline is allowed inside tabu and is the same as:

\firsthline \hline \hline.

```
1676 \def\tabu@firstline {\tabu@hlineAZ \tabu@firsthlinecorrection
1677 \def\tabu@firsthline{\tabu@hlineAZ \tabu@firsthlinecorrection \hline}
1678 \def\tabu@lastline {\tabu@hlineAZ \tabu@lasthlinecorrection
1679 \def\tabu@lasthline {\tabu@hlineAZ \tabu@lasthlinecorrection \hline}
1680 \def\tabu@hline {% replaces \hline if no colortbl (see \AtBeginDocument)
        \noalign{\ifnum0='}\fi
1681
1682
        {\CT@arc@\hrule height\arrayrulewidth}%
1683
        \futurelet \tabu@temp \tabu@xhline
1684 }% \tabu@hline
1685 \def\tabu@xhline{%
        \ifx \tabu@temp \hline
1686
1687
            {\ifx \CT@drsc@\relax \vskip
             \else\ifx \CT@drsc@\@empty \vskip
1688
1689
             \else \CT@drsc@\hrule height
1690
             \fi\fi
             \doublerulesep}%
1691
1692
        \fi
1693
        \ifnum0='{\fi}%
1694 }% \tabu@xhline
```

\tabu@hlineAZ abu@nexthlineAZ Here we go, inside a \noalign group, we collect the next tokens:

\tabu@nexthlineAZ \tabu@xhlineAZ

- 1. first the option,
- 2. and then the next tokens if they are \hline or \firsthline.

The code to be executed at the end of the \noalign group is built into \toks@.

```
1695 \def \align{\mbox{\count} \mbox{\count} \mbox{\coun
1696
                                  \label{linecorrection} $$ \c { } \def \tabu@temp{#2} % $$
1697
                                  \tabu@hlineAZsurround
1698 }% \tabu@hlineAZ
1699 \newcommand*\tabu@hlineAZsurround[1][\extratabsurround]{%
1700
                                  \extratabsurround #1\let\tabucline \tabucline@scan
                                                                                                   \tabu@hlinescan \let\firsthline \hline
1701
1702
                                 \let\cline
                                                                                                   \tabu@clinescan \let\lasthline \hline
1703
                                  \expandafter \futurelet \expandafter \tabu@temp
1704
                                                                                      \expandafter \tabu@nexthlineAZ \tabu@temp
1705 }% \tabu@hlineAZsurround
1706 \def\tabu@hlinescan
                                                                                                                {\tabu@thick \arrayrulewidth \tabu@xhlineAZ \hline}
```

```
1707 \def\tabu@clinescan #1{\tabu@thick \arrayrulewidth \tabu@xhlineAZ {\cline{#1}}}
1708 \def\tabucline@scan{\@testopt \tabucline@sc@n {}}
1709 \def \align{ } 1709 \def \align{ } 11 & \ali
1710 \def\tabu@nexthlineAZ{%
                            \ifx \tabu@temp\hline \else
1711
1712
                            \ifx \tabu@temp\cline \else
1713
                             \ifx \tabu@temp\tabucline \else
1714
                                               \tabu@hlinecorrection
1715
                            \fi\fi\fi
1716 }% \tabu@nexthlineAZ
1717 \def\tabu@xhlineAZ #1{%
1718
                            \toks@\expandafter{\the\toks@ #1}%
                            \@tempdimc \tabu@thick
                                                                                                                                                                                   % The last line width
1719
1720
                             \ifcase\count@ \@tempdimb \tabu@thick
                                                                                                                                                                                  % The first line width
1721
                            \else \advance\dimen@ \dimexpr \tabu@thick+\doublerulesep \relax
1722
                            \fi
                            \advance\count@ \@ne
                                                                                                                   \futurelet \tabu@temp \tabu@nexthlineAZ
1723
1724 }% \tabu@xhlineAZ
```

\tabu@firsthlinecorrection This is the "correction macro" for \firsthline, ie.a strut and a skip are inserted before the first \hline.

```
1725 \def \tabu@firsthlinecorrection{% \count@ = number of \hline -1}
1726
        \@tempdima \dimexpr \ht\@arstrutbox+\dimen@
1727
        \edef\firsthline{%
                                 <local in \noalign>
1728
            \omit \hbox to\z@{\hss{\noexpand\tabu@DBG{yellow}\vrule
1729
                        height \the\dimexpr\@tempdima+\extratabsurround
1730
                        depth \dp\@arstrutbox
1731
                        width \tabustrutrule}\hss}\cr
1732
            \noalign{\vskip -\the\dimexpr \@tempdima+\@tempdimb
1733
                                             +\dp\@arstrutbox \relax}%
1734
            \the\toks@
1735
       }\ifnum0='{\fi
                \expandafter}\firsthline % we are then !
1736
1737 }% \tabu@firsthlinecorrection
```

\tabu@lasthlinecorrection This is the "correction macro" for \lasthline, ie.a strut and a skip are inserted after the last \hline.

```
1738 \def\tabu@lasthlinecorrection{%
       \@tempdima \dimexpr \dp\@arstrutbox+\dimen@+\@tempdimb+\@tempdimc
1739
1740
       \edef\lasthline{%
                          <local in \noalign>
1741
           \the\toks@
1742
           \omit \hbox to\z@{\hss{\noexpand\tabu@DBG{yellow}\vrule
1743
1744
                       depth \the\dimexpr \dp\@arstrutbox+\@tempdimb+\dimen@
1745
                                         +\extratabsurround-\@tempdimc
1746
                       height \z@
1747
                       width \tabustrutrule}\hss}\cr
1748
       }\ifnum0='{\fi
1749
               \expandafter}\lasthline % we are then !
1750 }% \tabu@lasthlinecorrection
Allowing colored rules even if colortbl is not loaded.
1751 \def\tabu@LT@@hline{%
1752
       \ifx\LT@next\hline
1753
           \qlobal\let\LT@next \@qobble
```

\noalign{\penalty-\@medpenalty\vskip\doublerulesep}}%

\ifx \CT@drsc@\relax

\gdef\CT@LT@sep{%

1754 1755

1756

```
1757
            \else
1758
                 \gdef\CT@LT@sep{%
1759
                     \multispan\LT@cols{%
                     \CT@drsc@\leaders\hrule\@height\doublerulesep\hfill}\cr}%
1760
            \fi
1761
1762
        \else
            \global\let\LT@next\empty
1763
1764
            \gdef\CT@LT@sep{%
                 \noalign{\penalty-\@lowpenalty\vskip-\arrayrulewidth}}%
1765
1766
        \fi
1767
        \ifnum0='{\fi}%
1768
        \multispan\LT@cols
1769
            {\CT@arc@\leaders\hrule\@height\arrayrulewidth\hfill}\cr
1770
        \CT@LT@sep
        \multispan\LT@cols
1771
            {\CT@arc@\leaders\hrule\@height\arrayrulewidth\hfill}\cr
1772
1773
        \noalign{\penalty\@M}%
1774
        \I.T@next
1775 }% \tabu@LT@@hline
```

Horizontal lines: \tabucline

\tabucline \tabucline [style or spec.]{start-end}

\tabucline appears only at the end of a line: this is the place where we can insert a \noalign group. The line to be inserted inside the tabu is build inside this \noalign group.

\tabu@start and \tabu@stop store the limits for the line: they are, for clarity, the local name of \@tempcnta and \@tempcntb.

```
1776 \let\tabu@start \@tempcnta
1777 \let\tabu@stop \@tempcntb
1778 \newcommand*\tabucline{\noalign{\ifnum0='}\fi \tabu@cline}
1779 \newcommand*\tabu@cline[2][]{\tabu@startstop{#2}%
1780
       \ifnum \tabu@stop<\z@
                               \toks@{}%
1781
       \else \tabu@clinearg{#1}\tabu@thestyle
1782
            \edef\tabucline{\toks@{%
1783
                \ifnum \tabu@start>\z@ \omit
                      \tabu@multispan\tabu@start {\span\omit}&\fi
1784
                \omit \tabu@multispan\tabu@stop {\span\omit}%
1785
1786
                                              \tabu@thehline\cr
            }}\tabucline
1787
1788
            \tabu@tracinglines{(tabu:tabucline) Style: #1^^J\the\toks@^^J^^J}%
1789
        \fi
1790
        \futurelet \tabu@temp \tabu@xcline
1791 }% \tabu@cline
1792 \def\tabu@clinearg #1{%
        \ifx\ \tabu@thestyle \tabu@ls@
1793
1794
        \else \@defaultunits \expandafter\let\expandafter\@tempa
                                         \romannumeral-'\0#1\relax \@nnil
1795
1796
            \ifx \hbox\@tempa
                                         \tabu@clinebox{#1}%
1797
            \else\ifx \box\@tempa
                                         \tabu@clinebox{#1}%
1798
            \else\ifx \vbox\@tempa
                                         \tabu@clinebox{#1}%
            \else\ifx \vtop\@tempa
                                         \tabu@clinebox{#1}%
1799
1800
            \else\ifx \copy\@tempa
                                         \tabu@clinebox{#1}%
            \else\ifx \leaders\@tempa
                                         \tabu@clineleads{#1}%
1801
            \else\ifx \cleaders\@tempa
                                         \tabu@clineleads{#1}%
1802
1803
            \else\ifx \xleaders\@tempa
                                         \tabu@clineleads{#1}%
            \else\tabu@getline {#1}%
1804
            \fi\fi\fi\fi\fi\fi\fi
1805
```

\fi

1806

```
\mathsf{T}_{\aleph}b \subset [\text{rev.2.8 release}] \odot 2010 - 2011 \hookrightarrow \mathsf{FC}
```

\tabu@startstop

```
1807 }% \tabu@clinearg
1808 \def\tabu@clinebox #1{\tabu@clineleads{\xleaders#1\hss}}
1809 \def\tabu@clineleads #1{%
                   \let\tabu@thestyle \relax \let\tabu@leaders \@undefined
1810
1811
                   \gdef\tabu@thehrule{#1}}
1812 \def\tabu@thehline{\begingroup
                   \ifdefined\tabu@leaders
1814
                                       \noexpand\tabu@thehleaders
1815
                                       \noexpand\tabu@thehrule
                   \else
1816
                   \fi
                                                        \endgroup
1817 }% \tabu@thehline
1818 \def\tabu@xcline{%
1819
                   \ifx \tabu@temp\tabucline
1820
                             \toks@\expandafter{\the\toks@ \noalign
                             {\ifx\CT@drsc@\relax \vskip
1821
1822
                               \else \CT@drsc@\hrule height
1823
1824
                                \doublerulesep}}%
1825
                   \fi
                   \tabu@docline
1826
1827 }% \tabu@xcline
1828 \end{figure} $$1828 \end{figure} $$1828
1829 \def\tabu@docline@evr {\xdef\tabu@doclineafter{\the\toks@}%
                                            \ifnum0='{\fi}\aftergroup\tabu@doclineafter}
1831 \def\tabu@multispan #1#2{%
                   \ifnum\numexpr#1>\@ne #2\expandafter\tabu@multispan
1832
                                                                               \expandafter\@gobbletwo
1833
                   \else
1834
                   \fi {#1-1}{#2}%
1835 }% \tabu@multispan
This macro parses the mandatory argument of \tabucline: start-column and end-column of the \cline.
1836 \def\tabu@startstop #1{\tabu@start@stop #1\relax 1-\tabu@nbcols \@nnil}
1837 \det \theta = \#1-\#2\ensuremath{\%}
1838
                \@defaultunits
                                                          \tabu@start\number 0#1\relax
                                                                                                                                           \@nnil
                                                          \tabu@stop \number 0#2\relax
                                                                                                                                          \@nnil
1839
                \@defaultunits
1840
                \tabu@stop
                                                \ifnum \tabu@start>\tabu@nbcols
                                                                                                                                           \m@ne
1841
                                                 \left( \sum_{i=1}^{n} t_{i} \right)
                                                                                                                                           \tabu@nbcols
1842
                                                 \else\ifnum \tabu@stop>\tabu@nbcols \tabu@nbcols
1843
                                                 \else
                                                                                                                                           \tabu@stop
```

11.17 Numbers in tabu

1847 }% \tabu@start@stop

\tabudecimal

1844

1845

1846

\tabudecimal \tabudecimal is \tabudecimal inside the tabu environment.

\fi\fi\fi

\advance\tabu@start \m@ne

\ifnum \tabu@start>\z@ \advance\tabu@stop -\tabu@start \fi

```
\ifcase 0\ifx\tabu@temp\ignorespaces\else
1856
1857
                                                                                               \ifx\tabu@temp\@sptoken1\else
1858
                                                                                              2\fi\fi\relax
                                                                                         \let\tabu@getdecimal@ \tabu@getdecimal
1859
                                                                                         \expandafter\tabu@skipdecimal
1860
1861
                                           \or
                                                                                        \expandafter\tabu@gobblespace\expandafter\tabu@scandecimal
1862
                                           \else
                                                                                         \expandafter\tabu@skipdecimal
1863
                                           \fi
1864 }% \tabu@getdecimal@ignorespaces
1865 \ \texttt{\def} \ \texttt{\decimal} \ \#1{\texttt{\decimptokena}} \ \text{\end} \ 
1866
                                                                                                                                                           \tabu@scandecimal}
1867 \def\do#1{%
                                           \def\tabu@get@decimalspace#1{%
1869
                                                                   \@temptokena\expandafter{\the\@temptokena #1}\tabu@scandecimal}%
1870 }\do{ }
1871 \let\tabu@@tabudecimal \tabu@tabudecimal
```

\tabu@getdecimal

```
1872 \def \tabu@getdecimal{%}
1873
      \ifcase
                 0\ifx 0\tabu@temp\else
1874
                  \ifx 1\tabu@temp\else
1875
                   \ifx 2\tabu@temp\else
1876
                  \ifx 3\tabu@temp\else
1877
                  \ifx 4\tabu@temp\else
1878
                  \ifx 5\tabu@temp\else
                  \ifx 6\tabu@temp\else
1879
1880
                  \ifx 7\tabu@temp\else
1881
                  \ifx 8\tabu@temp\else
1882
                  \ifx 9\tabu@temp\else
1883
                  \ifx .\tabu@temp\else
                  \ifx ,\tabu@temp\else
1884
                  \  \ -\tabu@temp\else
1885
1886
                  \ifx +\tabu@temp\else
                   \ifx e\tabu@temp\else
1887
1888
                  \ifx E\tabu@temp\else
                  \ifx\tabu@cellleft\tabu@temp1\else
1889
1890
                  \ifx\ignorespaces\tabu@temp1\else
1891
                   \ifx\@sptoken\tabu@temp2\else
               1892
          \expandafter\tabu@get@decimal
1893
1894
      \or \expandafter\tabu@skipdecimal
      \or \expandafter\tabu@get@decimalspace
1895
1896
      \else\expandafter\tabu@printdecimal
1897
      \fi
1898 }% \tabu@getdecimal
1899 \def\tabu@printdecimal{%
1900
       \edef\tabu@temp{\the\@temptokena}%
1901
       \ifx\tabu@temp\@empty\else
       \ifx\tabu@temp\space\else
1902
1903
           \expandafter\tabu@decimal\expandafter{\the\@temptokena}%
       \fi\fi
1904
1905 }% \tabu@printdecimal
```

11.18 Verbatim inside tabu with X columns

\tabu@verbatim Setup to be done before \scantokens to allow verbatim inside the tabu environment.

```
1906 \def\tabu@verbatim{%
1907  \let\verb \tabu@verb
1908  \let\FV@DefineCheckEnd \tabu@FV@DefineCheckEnd
1909 }% \tabu@verbatim
```

Compatibility with LaTeX's kernel \verb command

\tabu@verb

The \verb macro from the latex kernel expands \@ifstar in a context where the space token: \Box has a category code of 12.

This is not compatible with \scantokens since \scantokens adds a space after each control sequence, including \verb:

```
\verb +some verbatim text+ becomes:
```

 $\ensuremath{\mbox{ }}$ \verb \upprox +some verbatim text+

and thus, the space token \Box is set as the \verb delimiter.

We therefore use (a silly) \@ifstar in order to gobble the possible space token.

```
1910 \label{tabu@ltx@verb \verb} \\ 1911 \def \tabu@verb{\@ifstar {\tabu@ltx@verb*} \tabu@ltx@verb}
```

Compatibility with the fancyvrb package

\tabu@FV@DefineCheckEnd This is quite the same issue as for LATEX \verb command: a space is inserted after each control sequence scanned by \scantoken.

This leads to a break in the macro that checks the end of a **Verbatim** environment, since this macro basically checks for a line that conforms to the pattern:

#1\end {#2}#3

while with \scantokens, such a line becomes:

#1\end | {#2}#3

in a context where the space token is not of category 10 (space).

Thus we replace the end-check for the Verbatim environment by a check on the detokenized-line (with ε -TEX \detokenize):

```
1912 \def\tabu@fancyvrb {%
1913
       \def\tabu@FV@DefineCheckEnd ##1{%
1914
            \def\tabu@FV@DefineCheckEnd{%
                ##1% <original definition (if fancyvrb is loaded)>
1915
                \let\FV@CheckEnd \tabu@FV@CheckEnd
1916
1917
                \let\FV@@CheckEnd
                                      \tabu@FV@@CheckEnd
1918
                \let\FV@@@CheckEnd \tabu@FV@@@CheckEnd
1919
                \edef\FV@EndScanning{%
1920
                \def\noexpand\next{\noexpand\end{\FV@EnvironName}}%
                    \global\let\noexpand\FV@EnvironName\relax
1921
1922
                    \noexpand\next}%
1923
                \xdef\FV@EnvironName{\detokenize\expandafter{\FV@EnvironName}}}}%
1924
        }\expandafter\tabu@FV@DefineCheckEnd\expandafter{\FV@DefineCheckEnd}
1925 }% \tabu@fancyvrb
1926 \def\tabu@FV@CheckEnd #1{\expandafter\FV@@CheckEnd \detokenize{#1\end{}}\@nil}
1927 \edef\tabu@FV@@@CheckEnd {\detokenize{\end{}}}
1928 \begingroup
1929 \catcode '\[1
                      \catcode '\]2
1930 \@makeother\{
                      \@makeother\}
1931
      \edef\x[\endgroup
```

```
1932 \def\noexpand\tabu@FV@@CheckEnd ##1\detokenize[\end{]##2\detokenize[}]##3%  
1933 ]\x \@ni1{\def\@tempa{#2}\def\@tempb{#3}}
```

\tabu@FV@ListProcessLine This macro replaces \FV@ListProcessLine when measuring the natural width of a Verbatim environment (see \tabu@startpboxmeasure)

```
1934 \def\tabu@FV@ListProcessLine #1{%
1935
      \hbox {%to \hsize{%
1936
        \kern\leftmargin
1937
        \hbox {%to \linewidth{%
1938
          \FV@LeftListNumber
1939
          \FV@LeftListFrame
1940
          \FancyVerbFormatLine{#1}\hss
1941 %% DG/SR modification begin - Jan. 28, 1998 (for numbers=right add-on)
            \FV@RightListFrame}%
1942 %%
1943
          \FV@RightListFrame
1944
          \FV@RightListNumber}%
1945 %% DG/SR modification end
1946
        \hss}}
```

11.19 \savetabu

\savetabu When this command is called by the user, the tabu preamble and target are globally stored into a macro \tabu@saved@\(user-name\).

```
1947 \newcommand*\savetabu[1] {\noalign{%
1948
       \tabu@sanitizearg{#1}\tabu@temp
       \ifx \tabu@temp\@empty \tabu@savewarn{}{The tabu will not be saved}\else
1949
1950
           \@ifundefined{tabu@saved@\tabu@temp}{}{\tabu@savewarn{#1}{Overwritting}}%
1951
           \ifdefined\tabu@restored \expandafter\let
               \csname tabu@saved@\tabu@temp \endcsname \tabu@restored
1952
1953
           \else {\tabu@save}%
           \fi
1954
       \fi}%
1955
1956 }% \savetabu
1957 \def\tabu@save {%
1958
       \toks0\expandafter{\tabu@saved@}%
1959
       \iftabu@negcoef
1960
           \let\tabu@wddef \relax \let\tabu@ \tabu@savewd \edef\tabu@savewd{\tabu@Xcoefs}%
           1961
1962
       \toks1\expandafter{\tabu@savedpream}%
1963
       \toks2\expandafter{\tabu@savedpreamble}%
1964
       \let\@preamble \relax
1965
       \let\tabu@savedpream \relax \let\tabu@savedparams \relax
1966
       \edef\tabu@preamble{%
1967
           \def\noexpand\tabu@aligndefault{\tabu@align}%
1968
           \def\tabu@savedparams {\noexpand\the\toks0}%
           \def\tabu@savedpream {\noexpand\the\toks1}}%
1969
1970
        \edef\tabu@usetabu{%
1971
           \def\@preamble {\noexpand\the\toks2}%
           \tabu@target \the\tabu@target \relax
1972
1973
           \tabucolX
                        \the\tabucolX
                                          \relax
1974
           \tabu@nbcols \the\tabu@nbcols \relax
           \def\noexpand\tabu@aligndefault{\tabu@align}%
1975
1976
           \def\tabu@savedparams {\noexpand\the\toks0}%
           \def\tabu@savedpream {\noexpand\the\toks1}}%
1977
       \let\tabu@aligndefault \relax \let\@sharp \relax
1978
1979
       \edef\@tempa{\noexpand\tabu@s@ved
1980
                              {\tabu@usetabu}
1981
                             {\tabu@preamble}
```

```
1982
                               {\the\toks1}}\@tempa
1983
       \tabu@message@save
1984 }% \tabu@save
1985 \long\def\tabu@s@ved #1#2#3{%}
1986
       \def\tabu@usetabu{#1}% <for \tabu@message@save>
1987
       \expandafter\qdef\csname tabu@saved@\tabu@temp\endcsname ##1{%
1988
           \ifodd ##1%
                           \usetabu
               \tabu@measuringfalse \tabu@spreadfalse % Just in case...
1989
1990
               \gdef\tabu@usetabu {%
                   \ifdim \tabu@target>\z@ \tabu@warn@usetabu \fi
1991
                   \global\let\tabu@usetabu \@undefined
1992
1993
                   \def\@halignto {to\tabu@target}%
                   #1%
1994
1995
                   \ifx \tabu@align\tabu@aligndefault@text
1996
                   \ifnum \tabu@nested=\z@
                          \let\tabu@align \tabu@aligndefault \fi\fi}%
1997
1998
           \else
                           \preamble
1999
               \qdef\tabu@preamble {%
                   \global\let\tabu@preamble \@undefined
2000
                   #2%
2001
                   \ifx \tabu@align\tabu@aligndefault@text
2002
                   \ifnum \tabu@nested=\z@
2003
2004
                          \let\tabu@align \tabu@aligndefault \fi\fi}%
2005
           \fi
           #3}%
2006
2007 }% \tabu@s@ved
2008 \def\tabu@aligndefault@text {\tabu@aligndefault}%
2009 \def\tabu@warn@usetabu {\PackageWarning{tabu}
2010
       {Specifying a target with \string\usetabu\space is useless
       \MessageBreak The target cannot be changed!}}
```

\tabu@savewarn

Info for overwritting when \savetabu is used.

\tabu@saveerr

Error if \usetabu is called with an unknown argument.

```
2013 \def\tabu@savewarn#1#2{\PackageInfo{tabu}}
2014 {User-name '#1' already used for \string\savetabu
2015 \MessageBreak #2}}%
2016 \def\tabu@saveerr#1{\PackageError{tabu}}
2017 {User-name '#1' is unknown for \string\usetabu
2018 \MessageBreak I cannot restore an unknown preamble!}\@ehd}
```

11.20 \rowfont

Setting font and alignment specification

\rowfont

\rowfont uses the control sequences \tabu@celllalign, \tabu@cellleft, \tabu@cellright and \tabu@cellralign which have been placed on purpose into the user-defined tokens inserted in any preamble by the array package.

\tabu@cellalign and \tabu@cellralign are used to modify the alignment. If the optional [alignment] parameter of \rowfont is not specified, then those control sequences expand to \@empty.

\tabu@cellleft contains the font-modification information.

Placement of those control sequences into the user-tokens that are inserted in the preamble by the array package is explained below under the macro \tabu@prepnext@tok.

```
2019 \newskip \tabu@cellskip
2020 \def\tabu@rowfont{\ifdim \baselineskip=\z@\noalign\fi
2021 {\ifnum0=`}\fi \tabu@row@font}
```

```
2022 \newcommand*\tabu@row@font[2][]{%
2023
        \ifnum7=\currentgrouptype
2024
            \global\let\tabu@@cellleft
                                           \tabu@cellleft
            \global\let\tabu@@cellright
                                           \tabu@cellright
2025
            \global\let\tabu@@celllalign \tabu@celllalign
2026
2027
            \global\let\tabu@@cellralign \tabu@cellralign
2028
            \global\let\tabu@@rowfontreset\tabu@rowfontreset
2029
       \fi
2030
        \global\let\tabu@rowfontreset \tabu@rowfont@reset
        \expandafter\gdef\expandafter\tabu@cellleft\expandafter{\tabu@cellleft #2}%
2031
        \ifcsname tabu@cell@#1\endcsname
2032
                                                % row alignment
2033
                \csname tabu@cell@#1\endcsname \fi
        \ifnum0='{\fi}% end of group / noalign group
2034
2035 }% \rowfont
2036 \def\tabu@ifcolorleavevmode #1{\let\color \tabu@leavevmodecolor #1\let\color\tabu@color}%
```

\tabu@rowfont@reset This macro restores \tabu@celllalign, \tabu@cellleft, \tabu@cellright, and \tabu@cellralign to the value they had before the expansion of \rowfont.

It expands when a new row is inserted into the tabular or array.

```
2037 \def\tabu@rowfont@reset {%
2038
       \global\let\tabu@rowfontreset \tabu@@rowfontreset
2039
        \global\let\tabu@cellleft
                                      \tabu@@cellleft
2040
       \global\let\tabu@cellright
                                      \tabu@@cellright
        \global\let\tabu@cellfont
2041
                                      \@empty
2042
       \global\let\tabu@celllalign \tabu@@celllalign
2043
       \global\let\tabu@cellralign \tabu@@cellralign
2044 }% \tabu@@rowfontreset
2045 \let\tabu@rowfontreset \@empty
                                       % overwritten \AtBeginDocument if colortbl
```

Preparing stuff to be able to use \rowfont

\tabu@prepnext@tok

\tabu@prepnext@tok will replace \prepnext@tok defined in array.sty (only inside a tabu environment). its purpose is to count the number of columns, and to insert the control sequences \tabu@celllalign, \tabu@cellleft, \tabu@cellright and \tabu@cellralign at the edge of each cell of the tabular. This is done by putting them inside the user-tokens placed around each column by the array package.

\prepnext@tok in array.sty initialises each user-token to an empty one, each time there is a need for a new one! The macro has a very simple definition, but it expansion is the occasion to look carefully at the counters \count@ and \@tempcnta which gives us all information required to decide is the token in preparation will be finally placed on the left or on the right of a column.

$$\underbrace{> \{ \texttt{\color \{red\}} \} }_{\texttt{\toks} < i>} r \underbrace{< \{ \texttt{\color \{black} \} \}, \ \ \ }_{\texttt{\toks} < i+1>}}$$

When a column is inserted in the tabular preamble (\P to T_EX counter count is equal to i+1 (ie.the right token) and the counter \P token is equal to i (ie.the left token). If the column is special (ie. \P or !) \P

Thus, when a new token is "prepared" by \prepnext@tok:

either: i =\count @ = @ tempcnta : the token to prepare (ie.\toks < i + 1 >) is the right one of a "normal" column. The switch \iftabu@cellright is set to true.

The previous token ($toks < i >= toks \setminus count@$) is necessarily the left one of this "normal" column: we prepend tabu@celllalign and append tabu@cellleft to this token (toks < i >). This token is finished and will not change afterwards.

or: $\mathbf{i} = \mathbf{0} = \mathbf{0} = \mathbf{0} = 1$: the token to prepare ($\mathbf{i} = \mathbf{0} = \mathbf{0}$) is either the left one of a normal column, or the single one of a special $\mathbf{0}$ or ! column.

If the switch \iftabu@cellright is true, then the *previous* token \toks < i > is the right one of the last inserted column (which was a "normal" column, thus):, \tabu@cellright \tabu@cellralign is appended to it, and the switch \ittabu@cellright is reset to false. May be \prepnext@tok will be

expanded again (by \save@decl): if it happens, then again \count@ =\@tempcnta +1 (same case) but \iftabu@cellright is false and nothing is changed.

else: The token to prepare (which is $\t < i + 1 >= \t < i + 1$), cannot be the right one of a "normal" column: $\t < i + 1 >= \t < i + 1$), cannot be the right one of a

The fact that $|\count@-\count@| > 1$ tells us that the previous token $\count@| > i$ is necessarily the single one of a "special" $\colone column$. We don't modify this token, as long as special columns are always inserted as is: $\colone column color has no effect on special columns, nor <math>\color column color has no effect on special columns, nor <math>\color column color has no effect on special columns, nor <math>\color column color column color has no effect on special columns, nor <math>\color column color column column color column column column column column color column co$

Thereafter, the original initialisation sequence occurs: $\advance \count@by\ene and initialize the token to prepare (<math>\count@=\count@initialize the token to prepare (<math>\count@initialize the token to prepare (\count@initialize the token t$

```
2046 \newif \iftabu@cellright
2047 \def\tabu@prepnext@tok{%
2048
        \ifnum \count@<\z@
                              % <first initialisation>
2049
                \@tempcnta \@M
                                  % <not initialized by array.sty>
2050
                \tabu@nbcols\z@
                \let\tabu@fornoopORI \@fornoop
2051
2052
                \tabu@cellrightfalse
2053
        \else
            \ifcase \numexpr \count@-\@tempcnta \relax % (case 0): prev. token is left
2054
2055
                     \advance \tabu@nbcols \@ne
                     \iftabu@cellright % before-previous token is right and is finished
2056
                         \tabu@cellrightfalse % <only once>
2057
2058
                         \tabu@righttok
                     \fi
2059
2060
                     \tabu@lefttok
2061
            \or
                                      % (case 1) previous token is right
2062
                     \tabu@cellrighttrue \let\@fornoop \tabu@lastnoop
            \else % special column: do not change the token
2063
                     \iftabu@cellright
                                         % before-previous token is right
2064
2065
                         \tabu@cellrightfalse
2066
                         \tabu@righttok
2067
                     \fi
            \fi % \ifcase
2068
        \fi
2069
        \tabu@prepnext@tokORI
2070
2071 }% \tabu@prepnext@tok
2072 \lceil \log \lceil \tan \lceil 2 \rceil \rceil + 2 \rceil \lceil 2 \rceil \rceil 
2073 \def\tabu@lastn@@p \#1\@nextchar \#2\#3\in@@{%
        ifx \in \#2\leq
2074
2075
            \let\@fornoop \tabu@fornoopORI
2076
            \xdef\tabu@mkpreambuffer{\tabu@nbcols\the\tabu@nbcols \tabu@mkpreambuffer}%
            \toks0\expandafter{\expandafter\tabu@everyrowtrue \the\toks0}%
2077
2078
            \expandafter\prepnext@tok
        \fi
2079
2080 }% \tabu@lastnoop
2081 \def\tabu@righttok{%
2082
        \advance \count@ \m@ne
2083
        \toks\count@\expandafter {\the\toks\count@ \tabu@cellright \tabu@cellralign}%
2084
        \advance \count@ \@ne
2085 }% \tabu@righttok
2086 \def\tabu@lefttok{\toks\count@\expandafter{\expandafter\tabu@celllalign
2087
                                          \the\toks\count@ \tabu@cellleft}% after because of $
2088 }% \tabu@lefttok
```

Neutralisation of glues and alignment modification

```
First initialisation to \@empty.
    \tabu@cellleft
\tabu@celllalign
                                    2089 \let\tabu@cellleft
                                                                                       \@empty
 \tabu@cellright
                                    2090 \let\tabu@cellright
                                                                                      \@empty
\tabu@cellralign
                                    2091 \tabu@celllalign@def{\tabu@cellleft}%
                                    2092 \let\tabu@cellralign \@empty
\tabu@cell@align
                                    2093 \det \text{tabu@cell@align } #1#2#3{%}
                                                    \let\tabu@maybesiunitx \toks@ \tabu@celllalign
                                                    \global \expandafter \tabu@celllalign@def \expandafter {\the\toks@ #1}%
                                    2095
                                    2096
                                                    \toks@\expandafter{\tabu@cellralign #2}%
                                                    \xdef\tabu@cellralign{\the\toks@} % \end{constraint} % \label{limit} % \end{constraint} % \end{constraint}
                                    2097
                                    2098
                                                    \toks@\expandafter{\tabu@cellleft #3}%
                                                    \xdef\tabu@cellleft{\the\toks@}%
                                    2099
                                    2100 }% \tabu@cell@align
                                    Setup macros to modify the alignment. The skips inserted to make the standard alignment specified in the
       \tabu@cell@1
                                    tabular preamble are not the same with standard array tabulars and colortbl tabulars, hence the switch
       \tabu@cell@c
                                    \iftabu@colortbl.
       \tabu@cell@r
        \tabu@cell@j
                                    2101 \def \tabu@cell@l{%} force alignment to left
                                    2102
                                                  \tabu@cell@align
                                    2103
                                                         {\tabu@removehfil \raggedright \tabu@cellleft}% left
                                    2104
                                                         {\tabu@flush1\tabu@ignorehfil}%
                                                                                                                                                           right
                                                         \raggedright
                                    2105
                                    2106 }% \tabu@cell@l
                                    2107 \def\tabu@cell@c{% force alignment to center
                                    2108
                                                  \tabu@cell@align
                                    2109
                                                         {\tabu@removehfil \centering \tabu@flush{.5}\tabu@cellleft}
                                    2110
                                                         {\tabu@flush{.5}\tabu@ignorehfil}
                                                         \centering
                                    2111
                                    2112 }% \tabu@cell@c
                                    2113 \def\tabu@cell@r{% force alignment to right
                                    2114
                                                  \tabu@cell@align
                                    2115
                                                         {\tabu@removehfil \raggedleft \tabu@flush1\tabu@cellleft}
                                    2116
                                                         \tabu@ignorehfil
                                                        \raggedleft
                                    2117
                                    2118 }% \tabu@cell@r
                                    2120
                                                        \tabu@cell@align
                                    2121
                                                               {\tabu@justify\tabu@cellleft}
                                    2122
                                    2123
                                                              \tabu@justify
                                    2124 }% \tabu@cell@j
                                    2125 \def\tabu@justify{%
                                    2126
                                                  \leftskip\z@skip \@rightskip\leftskip \rightskip\@rightskip
                                    2127
                                                  \parfillskip\@flushglue
                                    2128 }% \tabu@justify
                                    2129 %% ragged2e settings
                                    2130 \def\tabu@cell@L{% force alignment to left (ragged2e)
                                    2131
                                                  \tabu@cell@align
                                    2132
                                                         {\tabu@removehfil \RaggedRight \tabu@cellleft}
                                                         {\tabu@flush 1\tabu@ignorehfil}
                                    2133
                                                        \RaggedRight
                                    2134
                                    2135 }% \tabu@cell@L
                                    2136 \def\tabu@cell@C{% force alignment to center (ragged2e)
```

```
\mathcal{T}_{\aleph}b \subset [\text{rev.2.8 release}] \otimes 2010 - 2011 <math>\hookrightarrow \mathcal{F}
```

```
2137
       \tabu@cell@align
2138
          {\tabu@removehfil \Centering \tabu@flush{.5}\tabu@cellleft}
2139
          {\tabu@flush{.5}\tabu@ignorehfil}
          \Centering
2140
2141 }% \tabu@cell@C
2142 \def\tabu@cell@R{% force alignment to right (ragged2e)
2143
       \tabu@cell@align
2144
          {\tabu@removehfil \RaggedLeft \tabu@flush 1\tabu@cellleft}
          \tabu@ignorehfil
2145
2146
          \RaggedLeft
2147 }% \tabu@cell@R
2148 \def\tabu@cell@J{% force justification (ragged2e)
       \tabu@cell@align
2150
          {\justifying \tabu@cellleft}
2151
          { }
2152
          \justifying
2153 }% \tabu@cell@J
2154 \def\tabu@flush#1{%
        \iftabu@colortbl
                                % colortbl uses \hfill rather than \hfil
2155
            \hskip \ifnum13<\currentgrouptype \stretch{#1}%
2156
            \else \ifdim#1pt<\p@ \tabu@cellskip
2157
2158
            \else \stretch{#1}
2159
            \fi\fi \relax
2160
        \else
                               % arrav.stv
            \ifnum 13<\currentgrouptype
2161
                     \hfil \hskip1sp \relax \fi
2162
2163
        \fi
2164 }% \tabu@flush
```

\tabu@removehfil

\tabu@removehfil removes (eventually) the infinite stretchable glue inserted before the cell (in the preamble of \halign) to make the column alignment.

```
2165 \let\tabu@hfil \hfil
2166 \let\tabu@hfill \hfill
2167 \let\tabu@hskip \hskip
2168 \def\tabu@removehfil{%
2169
        \iftabu@colortbl
2170
            \unkern \tabu@cellskip =\lastskip
            \ifnum\qluestretchorder\tabu@cellskip =\tw@ \hskip-\tabu@cellskip
2171
2172
            \else \tabu@cellskip \z@skip
2173
            \fi
2174
        \else
            \ifdim\lastskip=1sp\unskip\fi
2175
            \ifnum\gluestretchorder\lastskip =\@ne
2176
                 \hfilneg % \hfilneg for array.sty but not for colortbl...
2177
2178
            \fi
2179
        \fi
2180 }% \tabu@removehfil
```

\tabu@ignorehfil

\tabu@ignorehfil removes (eventually) the infinite stretchable glue inserted after the cell (in the preamble of \halign) to make the column alignment.

```
2181 \def\tabu@ignorehfil{\aftergroup \tabu@nohfil}
2182 \def\tabu@nohfil{% \hfil -> do nothing + restore original \hfil
2183 \def\hfil{\let\hfil \tabu@hfil}% local to (alignment template) group
2184 }% \tabu@nohfil
2185 \def\tabu@colortblalignments {% if colortbl
2186 \def\tabu@nohfil{%
2187 \def\hfil {\let\hfil \tabu@hfil}% local to (alignment template) group
```

```
\def\hfill {\let\hfill \tabu@hfill}% (colortbl uses \hfill) pfff...
2188
            \def\hskip ####1\relax{\let\hskip \tabu@hskip}}% local
2190 }% \tabu@colortblalignments
```

11.21 Taking care of footnotes and \arraybackslash

Footnotes and hyperfootnotes

The macros in case hyperref is not used, or used with the option hyperfootnotes=false: \tabu@footenotetext

```
2191 \long\def\tabu@footnotetext #1{%
2192
       \edef\@tempa{\the\tabu@footnotes
2193
          \noexpand\footnotetext [\the\csname c@\@mpfn\endcsname]}%
2194
       \global\tabu@footnotes\expandafter{\@tempa {#1}}}%
2195 \long\def\tabu@xfootnotetext [#1]#2{%
2196
       \global\tabu@footnotes\expandafter{\the\tabu@footnotes
2197
                                    \footnotetext [{#1}]{#2}}}
2198 \let\tabu@xfootnote \@xfootnote
```

\tabu@Hy@ftntext

```
The macros in case hyperref is loaded with the option hyperfootnotes=true:
2200 \long\def\tabu@Hy@xfootnote [#1]{%
                  2201
                         \begingroup
                  2202
                            \wedge \mbox{\em walue}\em \mbox{\em mpfn $\#1$}
                  2203
                            \protected@xdef \@thefnmark {\thempfn}%
                  2204
                         \endgroup
                  2205
                         \@footnotemark \tabu@Hy@ftntxt {#1}%
                  2206 }% \tabu@Hy@xfootnote
                  2207 \long\def\tabu@Hy@ftntxt #1#2{%}
                  2208
                          \edef\@tempa{%
                  2209
                              \the\tabu@footnotes
                  2210
                              \begingroup
                  2211
                                  \value\@mpfn #1\relax
                  2212
                                  \noexpand\protected@xdef\noexpand\@thefnmark {\noexpand\thempfn}%
                  2213
                                  \expandafter \noexpand \expandafter
                  2214
                                      \tabu@Hy@footnotetext \expandafter{\Hy@footnote@currentHref}%
                  2215
                  2216
                          \global\tabu@footnotes\expandafter{\@tempa {#2}%
                  2217
                  2218 }% \tabu@Hy@ftntxt
                  2219 \long\def\tabu@Hy@footnotetext #1#2{%
                          \H@@footnotetext{%
                  2220
                  2221
                              \ifHy@nesting
                  2222
                                  \hyper@@anchor {#1}{#2}%
                  2223
                              \else
                  2224
                                  \Hy@raisedlink{%
                                      \hyper@@anchor {#1}{\relax}%
                  2225
                  2226
                                  } 응
                  2227
                                  \def\@currentHref {#1}%
                  2228
                                  \let\@currentlabelname \@empty
                  2229
                                  #2%
                              \fi
                  2230
                  2231
                          1 %
                  2232 }% \tabu@Hy@footnotetext
```

\centering, \raggedright, \raggedleft and \@normalcr

Inside tabu environment, no need to add \arraybackslash after such commands.

```
2233 \def\tabu@latextwoe {%
2234 \det \theta + 1#42#3{\{ toks@expandafter{##2##3} \times f#1{\the toks@} \}}
2235 \tabu@temp \tabu@centering \centering \arraybackslash
2236 \tabu@temp \tabu@raggedleft \raggedleft \arraybackslash
2237 \text{ } \text{tabu@temp } \text{tabu@raggedright } \text{arraybackslash}
2238 }% \tabu@latextwoe
2239 \def\tabu@raggedtwoe {%
2240 \eftabu@temp  ##1##2##3{{ \toks@}expandafter{##2##3} \xdef##1{\the\toks@}}}}
2241 \tabu@temp \tabu@Centering \Centering \arraybackslash
2242 \tabu@temp \tabu@RaggedLeft \RaggedLeft \arraybackslash
2244 \tabu@temp \tabu@justifying \justifying \arraybackslash
2245 }% \tabu@raggedtwoe
2246 \def\tabu@normalcrbackslash{\let\\\@normalcr}
2247 \def\tabu@trivlist{\expandafter\def\expandafter\@trivlist\expandafter{%
2248
                         \expandafter\tabu@normalcrbackslash \@trivlist}}
```

Utilities: tabu \fbox

\tabu@fbox works exactly like LATEX \fbox but allows the syntax: \fbox \bgroup...\egroup suitable for use inside tabular columns. \fbox is \let to \tabu@fbox at the entry inside a tabu environment.

```
2249 \def\tabu@fbox {\leavevmode\afterassignment\tabu@beginfbox \@tempboxa\hbox}
2250 \def\tabu@beginfbox {\bgroup \kern\fboxsep
2251 \bgroup\aftergroup\tabu@endfbox}
2252 \def\tabu@endfbox {\kern\fboxsep\egroup\egroup
2253 \@frameb@x\relax}
```

\tabu@fcolorbox works exactly like xcolor \fcolorbox but allows the syntax:

\fcolorbox {frame color}{background color}\bgroup...\egroup

suitable for use insed tabular columns. \fcolorbox is \let to \tabu@fcolorbox at the entry inside a tabu environment.

```
2254 \def\tabu@color@b@x #1#2{\leavevmode \bgroup
2255
      \afterassignment\tabu@begincolor@b@x \setbox\z@ \hbox
2256
2257 }% \tabu@color@b@x
2258 \def\tabu@begincolor@b@x {\kern\fboxsep \bgroup
         \aftergroup\tabu@endcolor@b@x \set@color}
2260 \def\tabu@endcolor@b@x {\kern\fboxsep \egroup
2261
      \dimen@\ht\z@ \advance\dimen@ \fboxsep \ht\z@ \dimen@
      \dimen@\dp\z@ \advance\dimen@ \fboxsep \dp\z@ \dimen@
2262
      \tabu@docolor@b@x \egroup
2263
2264 }% \tabu@endcolor@b@x
```

11.22 Corrections

delarray comptability fix for colortbl and arydshln

Both colortbl and arydshln forgot the control sequence \@arrayright which must be expanded by \endarray. Originally defined for delarray, this control sequence is used by tabu environments when tabu X columns are present in the preamble.

Here is the fix. We test if \endarray contains \@arrayright before modifying the control sequence, in case colortbl and/or arydshln modify their definition.

```
2265 \verb|\def| tabu@fix@arrayright {%% | @arrayright is missing from | endarrayright | } \\
```

```
2266
        \iftabu@colortbl
2267
            \ifdefined\adl@array % <colortbl + arydshln>
2268
            \def\tabu@endarray{%
                \adl@endarray \egroup \adl@arrayrestore \CT@end \egroup %<original>
2269
                                   % <FC>
2270
                \@arrayright
2271
                \gdef\@preamble{}}% <FC>
2272
            \else
                                   % <colortbl / no arydshln>
2273
            \def\tabu@endarray{%
2274
                \crcr \egroup \egroup
                                           %<original>
                \@arrayright
                                           % <FC>
2275
                \gdef\@preamble{}\CT@end}%
2276
2277
            \fi
2278
        \else
2279
            \ifdefined\adl@array % <arydshln / no colortbl>
2280
            \def\tabu@endarray{%
                \adl@endarray \egroup \adl@arrayrestore \egroup %<original>
2281
                \@arrayright
                                   % <FC>
2282
2283
                \qdef\@preamble{}}% <FC>
                                 % <no arydshln / no colotbl + \@arrayright missing>
2284
        \else
2285
            \PackageWarning{tabu}
            {\string\@arrayright\space is missing from the
2286
            \MessageBreak definition of \string\endarray.
2287
2288
            \MessageBreak Comptability with delarray.sty is broken.}%
2289
        \fi\fi
2290 }% \tabu@fix@arrayright
```

arydshln @ columns

```
2291 \def\tabu@adl@xarraydashrule #1#2#3{%
2292
         \ifnum\@lastchclass=\adl@class@start\else
2293
         \ifnum\@lastchclass=\@ne\else
2294
         \ifnum\@lastchclass=5 \else % <FC> @-arg (class 5) and !-arg (class 1)
                                                       % must be treated the same
                 \adl@leftrulefalse \fi\fi
2295
         \fi
2296
         \ifadl@zwvrule\else \ifadl@inactive\else
2297
2298
                 \@addtopreamble{\vrule\@width\arrayrulewidth
                          \@height\z@ \@depth\z@}\fi \fi
2299
2300
         \ifadl@leftrule
                 \@addtopreamble{\adl@vlineL{\CT@arc@}{\adl@dashgapcolor}%
2301
2302
                          {\number#1}#3}%
                 \@addtopreamble{\adl@vlineR{\CT@arc@}{\adl@dashgapcolor}%
2303
         \else
2304
                          {\number#2}#3}
2305
         \fi
2306 }% \tabu@adl@xarraydashrule
```

arydshln, colors without colortbl and empty p columns

arydshln redefines \@endpbox for p columns. The definition is stored in \adl@act@endpbox. Here it is:

```
\unskip \ifhmode \nobreak
```

```
\vrule\@width\z@\@height\z@\@depth\dp\@arstrutbox
```

\egroup \adl@colhtdp \box\adl@box \hfil

The \vrule inserted is exactly what package array calls: \@finalstrut \@arstrutbox.

However, just like in array.sty, this array-strut should be inserted inconditionnally, and \ifhmode applies only to \nobreak (misplaced \fi in arydshln definition).

Finally, arydshln is not compatible with colors in columns, such that: >{\color {red}}p3in, Unless colortbl is also loaded, the color group is missing.

Fixed inside tabu environment.

```
2307 \def\tabu@adl@act@endpbox {%
       \unskip \ifhmode \nobreak \fi
2308
                                         \@finalstrut \@arstrutbox
       \egroup \egroup
2309
2310
       \adl@colhtdp \box\adl@box \hfil
2311 }% \tabu@adl@act@endpbox
2312 \def\tabu@adl@fix {%
2313
       \let\adl@xarraydashrule \tabu@adl@xarraydashrule % <fix> arydshln
       \let\adl@act@endpbox
                               \tabu@adl@act@endpbox % <fix> arydshln
2314
2315
       \let\adl@act@@endpbox \tabu@adl@act@endpbox % <fix> arydshln
                                                          % <fix> arydshln
2316
       \let\@preamerror
                                \@preamerr
2317 }% \tabu@adl@fix
```

longtable \@startpbox: \everypar needed

\tabu@LT@startpbox The leading strut should be inserted at \everypar in order for \tabulinesep to work (otherwise, TFX is in horizontal mode and \nointerlineskip breaks).

```
2318 \def\tabu@LT@startpbox #1{%
2319
        \bgroup
2320
            \let\@footnotetext\LT@p@ftntext
2321
            \setlength\hsize{#1}%
2322
            \@arrayparboxrestore
2323
            \everypar{%
2324
                \vrule \@height \ht\@arstrutbox \@width \z@
2325
                \everypar{}}%
2326 }% \tabu@LT@startpbox
```

11.23 Package options and Initialisation

\tracingtabu and the package options

```
The delarray package option is only there for convenience: it simply loads the delarray package.
delarray (package option)
                    2327 \DeclareOption{delarray}{\AtEndOfPackage{\RequirePackage{delarray}}}}
```

The linegoal package option only sets \tabudefaulttarget to be equal to \linegoal. The required linegoal (package option) package linegoal is loaded.

```
2328 \DeclareOption{linegoal}{%
2329
       \AtEndOfPackage{%
2330
          \RequirePackage{linegoal}[2010/12/07]%
          \let\tabudefaulttarget \linegoal% \linegoal is \linewidth if not pdfTeX
2331
2332 }}
```

\tabu@@DBG

The scantokens package option makes tabu equal to tabu*. \scantokens (package option)

2333 \DeclareOption{scantokens} {\tabuscantokenstrue}

\tracingtabu is the same as the package option debugshow. \tracingtabu

```
 \begin{tabular}{ll} \textbf{debugshow} & \textbf{(package 2984 \beclareOption{debugshow}{\AtEndOfPackage{\tracingtabu=\tw@}}} & \textbf{(package 2000)} & \textbf{(pack
                                                                                                                              2335 \def\tracingtabu {\begingroup\@ifnextchar=%
                                                                                                                                                                                   {\afterassignment\tabu@tracing\count@}
              \mathsf{T}_{\aleph}b \subset [\text{rev.2.8 release}] \otimes 2010 - 2011 \hookrightarrow \mathsf{FC}
                                                                                                                              2336
                                                                                                                              2337
                                                                                                                                                                                    {\afterassignment\tabu@tracing\count@1\relax}}
                                                                                                                              2338 \def\tabu@tracing{\expandafter\endgroup
                                                                                                                                                                                   \expandafter\tabu@tr@cing \the\count@ \relax
                                                                                                                              2340 }% \tabu@tracing
                                                                                                                              2341 \def\tabu@tr@cing #1\relax {%
                                                                                                                                                                                 \ifnum#1>\thr@@ \let\tabu@tracinglines\message
                                                                                                                              2342
                                                                                                                                                                                                                                                                                               \let\tabu@tracinglines\@gobble
                                                                                                                              2343
                                                                                                                                                                                  \else
                                                                                                                              2344
                                                                                                                                                                                   \fi
```

\let\tabu@DBG

\ifnum#1>\tw@

2345

```
2346
                          \def\tabu@mkarstrut {\tabu@DBG@arstrut}%
2347
                          \tabustrutrule
                                               1.5\p@
2348
        \else
                          \let\tabu@DBG
                                                \@gobble
                          \def\tabu@mkarstrut {\tabu@arstrut}%
2349
2350
                          \tabustrutrule
                                               120
2351
        \fi
2352
        \ifnum#1>\@ne
                         \let\tabu@debug
                                                \message
2353
        \else
                         \let\tabu@debug
                                                \@gobble
        \fi
2354
        \infnum#1>\z@
2355
            \let\tabu@message
2356
                                             \message
2357
            \let\tabu@tracing@save
                                             \tabu@message@save
2358
            \let\tabu@starttimer
                                             \tabu@pdftimer
2359
        \else
2360
            \let\tabu@message
                                             \@gobble
            \let\tabu@tracing@save
                                             \@gobble
2361
2362
            \let\tabu@starttimer
                                             \relax
2363
        \fi
2364 }% \tabu@tr@cing
```

Initialisation and setup \AtBeginDocument

At the end of the tabu package:

- \tracingtabu is set to 0: this initialises the message commands. Eventually,, t he value will be overwritten by the debugshow package option later.
- \everycow is set to empty: this initialises the process at \everycr to the default process,
- a new *empty* line style is defined, to be equivalent to **\hline**: this creates the *default leaders*, which will be used if a line style specification cannot be parsed successfully.

 Then this default line style is set to be the current one.

At Begin Document, a fix for arydshln and colortbl comptability with delarray shortcuts available inside tabu: requirement for this fix is checked by \tabu@fix@arrayright.

Then the switch \iftabu@colortbl is set.

Finally, the longtabu environment is defined only if the longtable package is detected.

```
2365 \AtBeginDocument { \tabu@AtBeginDocument }
2366 \def\tabu@AtBeginDocument {\let\tabu@AtBeginDocument \@undefined
2367
        \ifdefined\arrayrulecolor
                                     \tabu@colortbltrue
                                                               % <colortbl>
                                     \tabu@colortblalignments % different glues are used
2368
2369
        \else
                                     \tabu@colortblfalse \fi
2370
        \ifdefined\CT@arc@ \else \let\CT@arc@ \relax \fi
        \ifdefined\CT@drsc@\else \let\CT@drsc@ \relax \fi
2371
2372
        \let\tabu@arc@L \CT@arc@ \let\tabu@drsc@L \CT@drsc@
2373
        \ifodd 1\ifcsname siunitx_table_collect_begin:Nn\endcsname
                                                                        % <siunitx: ok>
2374
                \expandafter\ifx
2375
                    \csname siunitx_table_collect_begin:Nn\endcsname\relax 0\fi\fi\relax
                \tabu@siunitxtrue
2376
                \let\tabu@maybesiunitx
                                          \@firstofone
2377
        \else
                                                                        % <not siunitx: setup>
                                          \tabu@nosiunitx
2378
                \let\tabu@siunitx
2379
                \tabu@siunitxfalse
2380
2381
        \ifdefined\adl@array
                                     % <arydshln>
                  \let\tabu@adl@fix \relax
2382
2383
                  \let\tabu@adl@endtrial \@empty \fi
2384
        \ifdefined\longtable
                                     % <longtable>
2385
                  \let\longtabu \tabu@nolongtabu \fi
2386
        \ifdefined\cellspacetoplimit \tabu@warn@cellspace\fi
```

```
\csname\ifcsname ifHy@hyperfootnotes\endcsname % <hyperfootnotes>
2387
2388
                ifHy@hyperfootnotes\else iffalse\fi\endcsname
2389
            \let\tabu@footnotetext \tabu@Hy@ftntext
                                    \tabu@Hy@xfootnote \fi
2390
            \let\tabu@xfootnote
        \ifdefined\FV@DefineCheckEnd% <fancyvrb>
2391
2392
                \tabu@fancyvrb \fi
2393
        \ifdefined\color
                                     % <color / xcolor>
2394
            \let\tabu@color \color
2395
            \def\tabu@leavevmodecolor ##1{%
2396
                \def\tabu@leavevmodecolor {\leavevmode ##1}%
2397
            }\expandafter\tabu@leavevmodecolor\expandafter{\color}%
2398
        \else
2399
            \let\tabu@color
                                       \tabu@nocolor
2400
            \let\tabu@leavevmodecolor \@firstofone \fi
2401
        \tabu@latextwoe
2402
        \ifdefined\@raggedtwoe@everyselectfont
                                                  % <ragged2e>
2403
            \tabu@raggedtwoe
2404
        \else
2405
            \let\tabu@cell@L \tabu@cell@l
2406
            \let\tabu@cell@R \tabu@cell@r
2407
            \let\tabu@cell@C \tabu@cell@c
2408
            \let\tabu@cell@J \tabu@cell@j
                                             \fi
2409
        \expandafter\in@ \expandafter\@arrayright \expandafter{\endarray}%
2410
        \ifin@ \let\tabu@endarray \endarray
2411
        \else \tabu@fix@arrayright \fi% <fix for colortbl & arydshln (delarray)>
2412
        \everyrow{}%
2413 }% \tabu@AtBeginDocument
2414 \def\tabu@warn@cellspace{%
2415
        \PackageWarning{tabu}{%
2416
                      Package cellspace has some limitations
2417
        \MessageBreak And redefines some macros of array.sty.
2418
        \MessageBreak Please use \string\tabulinesep\space to control
2419
        \MessageBreak vertical spacing of lines inside tabu environnement}%
2420 }% \tabu@warn@cellspace
```

\ProcessOption * is much quicker than without the star...

```
2421 \tabuscantokensfalse
2422 \let\tabu@arc@G
                              \relax
2423 \let\tabu@drsc@G
                              \relax
2424 \let\tabu@evr@G
                              \@empty
2425 \let\tabu@rc@G
                              \@empty
2426 \def\tabu@ls@G
                              {\tabu@linestyle@}%
2427 \let\tabu@@rowfontreset \@empty % <init>
2428 \let\tabu@@celllalign
                            \@empty
2429 \let\tabu@@cellralign
                             \@empty
2430 \let\tabu@@cellleft
                             \@empty
2431 \let\tabu@@cellright
                              \@empty
2432 \def\tabu@naturalXmin
                             {\z@}
2433 \def\tabu@naturalXmax
                              \{ \z @ \}
2434 \let\tabu@rowfontreset \@empty
2435 \def \tabulineon {4pt} \let \tabulineoff \tabulineon
2436 \tabu@everyrowtrue
2437 \ \text{ifdefined} \ \text{pdfelapsedtime}
                                                    % <pdfTeX>
            \def\tabu@pdftimer {\xdef\tabu@starttime{\the\pdfelapsedtime}}}
2438
            \let\tabu@pdftimer \relax \let\tabu@message@etime \relax
2439 \else
2440 \fi
2441 \tracingtabu=\z@
2442 \newtabulinestyle {=\maxdimen}% creates the 'factory' settings \tabu@linestyle@
```

```
2443 \tabulinestyle{}
2444 \taburowcolors{}
2445 \let\tabudefaulttarget \linewidth
2446 \ProcessOptions* % \ProcessOptions* is quicker !
2447 \( / package \)
```

12 References

- [1] A new implementation of LaTeX's tabular and array environments by Frank Mittelbach 2008/09/09 v2.4c Tabular extension package (FMi)

 CTAN:help/Catalogue/entries/array.html
- [2] The varwidth package by Donald Arseneau 2009/03/30 ver 0.92 Variable-width minipages CTAN:help/Catalogue/entries/varwidth.html
- [3] The enumitem-zref package by C 2011/02/18 ver 1.8 Extended references for enumitem pkg CTAN:help/Catalogue/entries/enumitem-zref.html

13 History

[2011/02/26 v2.8]

• Bug in the starred version (with \scantokens) of the longtabu* environment.

[2011/02/25 v2.7]

- Automatic \par after the end of the tabu environment used with its default target is removed in case of tabu spread: this was a bug.
- Some \ignorespaces were missing (in \everyrow, \taburulecolor, \taburowcolors and \tabulinestyle).

[2011/02/24 v2.6]

- \savetabu now also saves \tabulinesep (ie.\abovetabulinesep and \belowtabulinesep)
- Bug fixed for custom-environments when nested.
- \taburulecolor works even if colortbl is not loaded for the tabu environment.

 This is now the same for the longtabu environment.

[2011/02/19 v2.5]

- Bug fixed for \pdfelapsedtime when compilation without pdfT_FX.
- Modification of \@finalstrut ("null-rule" added) to avoid problems with \columncolor.

[2011/02/17 v2.4]

• Documentation revisited

[2011/02/13 v2.3]

• Fixed two bugs for nested tabu environment: when using \rowfont and when tabu is nested inside longtabu

[2011/02/12 v2.2 - New implementation - Absolutely no modification of array.sty]

• $\mathsf{T}_\aleph b \subset$ has been totally reimplemented, including the algorithms. In particular, outside of the tabu environment, absolutely none of the macros of array.sty, (and obviously none of LATEX) is modified.

The process has been completely reinvented: tabu follows a path along different modes (or strategies) measuring natural width of cells, fixing X column widths, measuring vertical length of rows and then printing the final tabular. The process is optimized, especially in the case of nested tabu environments: a tabular is not built twice for measuring purpose... As a result, many new features are now possible... vertical leaders (dashed lines), dynamic vertical spacing adjustment, and hopefully still more in a next release.

tabu now systematically collects the environment body. But with \scantokens, it is possible to insert verbatim material inside the columns: use tabu* instead of tabu, for the outer most tabular.

 $\mathsf{T}_{\aleph}b \subset [\text{rev.2.8 release}] \otimes 2010 - 2011 \Leftrightarrow \mathsf{FC}$

- New: \firsthline and \lasthline can draw multiple lines, and there is an option to set \extratabsurround instantly, and locally.
- New: \taburulecolor with a good behaviour with groupings (like \everyrow)
- Modification: \tabulinestyle sets the line style for the tabu, \newtabulinestyle defines a new line style.



[2011/01/19 v2.1]

- Vertical spacing had a bug with longtabu and paragraph columns. Fixed
- New: \everyrow.
- Fix a bug of \rowfont when using siunitx S columns.
- Some code optimisation.
- To do (if possible): a syntax X[6mc]S[...] to "embed" siunitx S column inside tabu and longtabu X columns...

[2011/01/18 v2.0]

- Vertical spacing of lines implemented! See \tabulinesep and \extrarowsep.
- \tabulinestyle: user defined line style can now be used inside the optional argument of the |[...] preamble token.
- [...] is now allowed in \multicolumn preamble inside tabu environment.
- Bug fixed inside \tabu@prepnexttok (again !!! a difficult case !)
- Incompatibility of package cellspace with tabu spread and tabu with negativ coefficients for X columns with has been lifted.

However, as said in the documentation of package cellspace, S column modifier does not work in the case of nested tabulars.

The S column modifier becomes C when the package siunitx is loaded (see siunitx documentation).

Moreover, cellspace does not work with color or xcolor and paragraph column types !! Finally, cellspace redefines globally \@startpbox and \@endpbox and is therefore not fully compatible with array.sty and therefore with $\mathcal{T}_{\aleph}b\subset$.

For all those reasons, $\tau_{\aleph}b \subset$ displays a warning to discourage the use of cellspace with the tabu environment.

[2011/01/15 v1.9]

- Bug in \savetabu when used inside longtabu...
- Bug when tabu with X column is nested inside lontabu.
- Documentation (\rowfont was missing in the summary).

[2010/12/28 v1.8]

- \tracingtabu / debugshow package option: reporting of the time elapsed during trials (if \pdfelapsedtime and thus pdfTEX is available) Slight modifications for better reporting on the .log file.
- Fix a bug when \savetabu is used after \multicolumn (\multicolumn globally redefines \@preamble).
- Fix a bug with \tabucline and \CT@arc@ (colortbl).
- Better privacy of columns types specifically defined for tabu.
- Improvement in the rewritting process (but only very few people should notice...)
- Documentation.

[2010/12/18 v1.7]

- Code optimisation
- Modification in the columns rewritting process (bug with some new column types defined by the user).

[2010/12/07 v1.5]

- Implementation of negative width coefficients for X columns (cf. tabu X columns Mastering horizontal space point 2).
- Columns natural widths computation (for tabu spread with X columns and negativ coefficients) is based on the code of the varwidth package by Donald Arseneau.
- longtabu is now provided, based on the longtable package by David Carlisle. longtabu can be used just like tabu.
- $\bullet\,$ Vertical lines can be used whatever the catcode of | is.
- \savetabu reports saved informations in the .log (debugshow option).
- \savetabu... \usetabu now restores the \halign preamble rather than the tabu preamble! \preamble can be use in the tabu preamble to restore a tabu preamble.
- \tabucline is more robust with "special" preambles containing > or < tokens. \tabucline now takes care of \arrayrulecolor (package colortbl).
- enumitem-zref package has been added to the documentation (see the link point 1)
- Optimisation of some parts of the code.

[2010/11/22 v1.4]

- Compatibility improvement with linegoal for the syntax: \begin {tabu} to\linegoal {...}
- Hyper footnotes now work correctly.
- Fix a bug when using colored vertical lines in tabu in math mode.
- Fix a bug with vertical lines and colortbl \arrayrulecolor specification.
- Fix a compatibility bug with arydshln: when nesting a tabular that use vertical dashed lines (arydshln) inside tabu spread with X columns.

[2010/11/18 v1.3]

- Fix a bug that may appear in \tabucline depending on the preamble due to arbitrary \countdef.
- Improvement in the use of \everycr: no \global stuff. Thus bug fixed when nesting tabu inside AMS-align environment for example. Same issue with \rowfont which now works without global modification of \everycr.
- No phantom line is added to tabu but a command \tabuphantomline is provided for this purpose (required with \multicolumn in some cases).
- Improvement on vertical alignment.
- To do: an example file to test a wide range of possibilities...
- Documentation.

[2010/11/15 v1.2]

- Improvement in parameters parsing for optional parameters (| and \tabucline).
- Modification / optimization in \tabu@prepnext@tok.
- Modification of \tabucline to get better results with m columns (X[m]) and also when \minrowclearance > 0 (package colortbl).

[2010/10/28 v1.1]

• First version.

14 Index

Numbers written in italic refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; plain numbers refer to the code lines where the entry is used.

${f Symbols}$	\@xhline 623
\& 80	\{
\@arrayparboxrestore 2322	\}
\@arrayright 1034, 1041, 1048,	
2265, 2270, 2275, 2282, 2284, 2286, 2409	A
\@arraystretch 693	\abovetabulinesep . <u>39, 200, 202, 204, 209,</u>
\@arstrutbox 109, 117, 118, 123, 624, 691, 692,	211, 213, 216, 217, 741, 1515, 1614, 1626
821, 822, 1003, 1284, 1555, 1638, 1654,	\adl@endarray 2269, 2281
1726, 1730, 1733, 1739, 1742, 1744, 2308, 2324	\afterassignment
\@defaultunits 330, 331, 858, 1063, 1794, 1838, 1839	173, 176, 204, 207, 238, 240, 241, 483,
\@endpbox 615, 638	575, 947, 1243, 2249, 2256, 2336, 2337
\@finalstrut 1036, 1274, 1555, 1631, 2308	\aftergroup
\@flushglue 1073, 1544, 2127	163, 164, 194, 195, 657, 660, 834, 996,
\@footnotemark 2205	997, 1243, 1546, 1830, 2181, 2251, 2259 \array 540
\@footnotetext 627, 1127, 2320	\array
\@gobbletwo 847, 1399, 1525, 1833	2235, 2236, 2237, 2241, 2242, 2243, 2244
\@halignto 873,	\arraycolsep
882, 1041, 1131, 1267, 1280, 1281, 1993	\arrayrulecolor
\@ifnextchar	\arrayrulewidth 429, 736, 1510,
168, 177, 197, 199, 208, 237, 285, 854, 2335	1682, 1706, 1707, 1765, 1769, 1772, 2298
\@mkpream 834	\arraystretch
\@normalcr	111, 115, 116, 126, 131, 743, 1517, 1649
\@onelevel@sanitize	, -, -, -, - ,,,
\@preamble 1077, 1081, 1085,	В
1092, 1502, 1964, 1971, 2271, 2276, 2283	${\bf B} \\ \verb+\baselineskip$
$1092,\ 1502,\ 1964,\ 1971,\ 2271,\ 2276,\ 2283$ \@raggedtwoe@everyselectfont 2402	_
1092, 1502, 1964, 1971, 2271, 2276, 2283 \@raggedtwoe@everyselectfont 2402 \@rightskip	\baselineskip
1092, 1502, 1964, 1971, 2271, 2276, 2283 \@raggedtwoe@everyselectfont	\baselineskip
1092, 1502, 1964, 1971, 2271, 2276, 2283 \@raggedtwoe@everyselectfont	\baselineskip
1092, 1502, 1964, 1971, 2271, 2276, 2283 \@raggedtwoe@everyselectfont	\baselineskip
1092, 1502, 1964, 1971, 2271, 2276, 2283 \@raggedtwoe@everyselectfont	$\begin & & & & & & & & & \\ \begin & & & & & & & & & \\ \belowtabulinesep & & & & & & & & \\ \begin & & & & & & & & \\ \belowtabulinesep & & & & & & & \\ \begin & & & & & & & \\ \begin & & & & & & & \\ \begin & & & & & & & \\ \begin & & & & \\ \begin & & & & & \\ \begin & & \\ \begin & & \\ \begin & & \\ \begin & & $
1092, 1502, 1964, 1971, 2271, 2276, 2283 \@raggedtwoe@everyselectfont	$\begin & & & & & & & & \\ begin & & & & & & & & \\ belowtabulinesep & & & & & & & \\ 211, 213, 216, 217, 742, 1516, 1615, 1627 \\ & & & & & \\ \hline C \\ c@footnote & & & & & \\ c@page & & & & & \\ 1422, 1424 \\ \endalignment$
1092, 1502, 1964, 1971, 2271, 2276, 2283 \@raggedtwoe@everyselectfont	$\begin & 2020 \\ begin & 8, 9 \\ belowtabulinesep & 39, 200, 202, 209, \\ & 211, 213, 216, 217, 742, 1516, 1615, 1627 \\ \hline \textbf{C} \\ \\ c@footnote & 2199 \\ \\ c@page & 1422, 1424 \\ \\ c@taburow & 21, 254, 349, 351, \\ \end{tabular}$
1092, 1502, 1964, 1971, 2271, 2276, 2283 \@raggedtwoe@everyselectfont	$\begin & 2020 \\ begin & 8, 9 \\ belowtabulinesep & 39, 200, 202, 209, \\ & 211, 213, 216, 217, 742, 1516, 1615, 1627 \\ \hline \textbf{C} \\ \\ c@footnote & 2199 \\ \\ c@page & 1422, 1424 \\ \\ c@taburow & 21, 254, 349, 351, \\ & 352, 369, 373, 685, 721, 758, 1636, 1637, \\ \end{tabular}$
$\begin{array}{cccccccccccccccccccccccccccccccccccc$	$\begin & & & & & & & & \\ begin & & & & & & & & \\ belowtabulinesep & & & & & & & \\ 211, 213, 216, 217, 742, 1516, 1615, 1627\\ \hline \\ \hline$
1092, 1502, 1964, 1971, 2271, 2276, 2283 \@raggedtwoe@everyselectfont	$\begin & & & & & & & & \\ begin & & & & & & & & \\ belowtabulinesep & & & & & & & \\ 211, 213, 216, 217, 742, 1516, 1615, 1627\\ \hline \\ \hline$
1092, 1502, 1964, 1971, 2271, 2276, 2283 \@raggedtwoe@everyselectfont	$\begin & & & & & & & & \\ begin & & & & & & & & & \\ belowtabulinesep & & & & & & & & \\ 211, 213, 216, 217, 742, 1516, 1615, 1627 \\ \hline \\ $
1092, 1502, 1964, 1971, 2271, 2276, 2283 \@raggedtwoe@everyselectfont	$\begin & & & & & & & & \\ begin & & & & & & & & & \\ belowtabulinesep & & & & & & & & \\ 211, 213, 216, 217, 742, 1516, 1615, 1627 \\ \hline \\ $
1092, 1502, 1964, 1971, 2271, 2276, 2283 \@raggedtwoe@everyselectfont	$\begin & & & & & & & & \\ begin & & & & & & & & & \\ belowtabulinesep & & & & & & & & \\ 211, 213, 216, 217, 742, 1516, 1615, 1627 \\ \hline \\ $
1092, 1502, 1964, 1971, 2271, 2276, 2283 \@raggedtwoe@everyselectfont 2402 \@rightskip 1978 \@sharp	$\begin & & & & & & & & \\ begin & & & & & & & & & \\ belowtabulinesep & & & & & & & & \\ 211, 213, 216, 217, 742, 1516, 1615, 1627 \\ \hline \\ $
1092, 1502, 1964, 1971, 2271, 2276, 2283 \@raggedtwoe@everyselectfont	$\begin & & & & & & & & \\ begin & & & & & & & & & \\ belowtabulinesep & & & & & & & & \\ 211, 213, 216, 217, 742, 1516, 1615, 1627 \\ \hline \\ $
1092, 1502, 1964, 1971, 2271, 2276, 2283 \@raggedtwoe@everyselectfont	$\begin & & & & & & & & & \\ begin & & & & & & & & & & & \\ belowtabulinesep & & & & & & & & & \\ 211, 213, 216, 217, 742, 1516, 1615, 1627 & & & & \\ \hline & & & & & \\ \hline & & & & & \\ \hline & & & &$
1092, 1502, 1964, 1971, 2271, 2276, 2283 \@raggedtwoe@everyselectfont	$\begin & & & & & & & & & & & \\ begin & & & & & & & & & & & & & & & \\ belowtabulinesep & & & & & & & & & & & & \\ 211, 213, 216, 217, 742, 1516, 1615, 1627 \\ \hline & & & & & & & \\ \hline & & & & & & \\ \hline & & & &$
1092, 1502, 1964, 1971, 2271, 2276, 2283 \@raggedtwoe@everyselectfont	$\begin & & & & & & & & & & & & \\ begin & & & & & & & & & & & & & & & & \\ belowtabulinesep & & & & & & & & & & & & \\ 211, 213, 216, 217, 742, 1516, 1615, 1627 & & & & & \\ \hline & & & & & & & & \\ \hline & & & &$
1092, 1502, 1964, 1971, 2271, 2276, 2283 \@raggedtwoe@everyselectfont	$\begin & & & & & & & & & & & & & & & & & & &$

$\verb \color@begingroup 1553$	\FV@LeftListNumber 1938
\color@block	\FV@ListProcessLine 1543
\CT@do@color	${f G}$
\CT@everycr	\GenericError 150, 1125
\CT@LT@sep 1755, 1758, 1764, 1770	\globaldefs 696, 723, 1139, 1653, 1668
\currentgrouplevel 225, 228, 229, 231	\gluestretchorder 2171, 2176
\currentgrouptype	
1600, 1605, 1606, 2023, 2156, 2161	Н
	\H@@footnotetext
D	\hbadness
\d@llarbegin 1603	\hfilneg
\d@llarend 136, 1613	\hline
δ debugshow δ (package δ option) $\underline{2334}$	1686, 1701, 1702, 1706, 1711, 1725, 1752
\definecolorseries	\hrule 439,
\delarray_(package_option) 2327	516, 1632, 1682, 1689, 1760, 1769, 1772, 1822
\detokenize	\ht 110, 115, 117, 126, 517, 691, 821, 822, 1614,
1521, 1522, 1523, 1923, 1926, 1927, 1932 \dimen@ 1142, 1143,	1626, 1638, 1647, 1726, 2255, 2261, 2324
1144, 1149, 1315, 1318, 1322, 1327, 1365,	
1367, 1373, 1383, 1384, 1409, 1429, 1695,	I
1721, 1726, 1739, 1742, 1744, 2261, 2262	\ifin@
\displaystyle 954	\iftabu@cellright 2046, 2056, 2064
\doublerulesep	\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
. 737, 1511, 1691, 1721, 1756, 1760, 1824	\iftabu@everyrow <u>49</u> , 103, 105, 163, 194, 255, 273, 297, 310, 339, 347, 641, 1245
\dp 111, 116, 118, 131,	\iftabu@long 49, 655, 996, 1162, 1215, 1254
692, 821, 822, 1615, 1627, 1638, 1648,	\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\
1730, 1733, 1739, 1742, 1744, 2255, 2262	583, 871, 1037, 1043, 1100, 1303, 1357, 1386
${f E}$	\iftabu@negcoef
\ecolumn	\iftabu@siunitx $\underline{49}$, 1006
\empty	\iftabu@spread 49, 594, 605, 874, 1200, 1298, 1534
\endarray 540, 640, 1163,	\iftabuscantokens
1223, 1254, 1289, 2265, 2287, 2409, 2410	<u>58</u> , 849, 1134, 1161, 1219, 1228, 1247
\endlinechar 100, 402, 1134	\ignorespaces
\endlongtable 550	\immediate
\endlongtabu 550, 553	\in@ 335, 337, 338, 2072, 2074, 2409
\endtabu	\in@false 427
\endtabular	\in@true 426
\everycr 614, 634, 1240, 1241, 1244	\indent 602
\everypar 554, 634, 646, 647, 1273, 1599, 2323, 2325	J
\everyrow 17, 234, 386, 729, 764, 2412	\justifying 632, 926, 2150, 2152, 2244
\extracolsep 1073	(Jassing 302, 023, 2133, 2132, 221
\extrarowdepth 39 , 111, 116, 131, 169, 171, 173,	${f L}$
178, 180, 182, 185, 186, 692, 740, 1514, 1648	\lastbox 648
\extrarowheight 110, 115, 126, 169, 171,	\lasthline 621, 1702, 1740, 1749
178, 180, 182, 185, 186, 691, 739, 1513, 1647	\lastline
\extrarowsep	\lccode
385, 738, 1512, 1699, 1700, 1729, 1745	\leaders 439, 1760, 1769, 1772, 1801
1. 300, 100, 1012, 1000, 1100, 1120, 1110	\leftmargin
${f F}$	\leftskip
\firsthline 15, 621, 1701, 1727, 1736	\linegoal 1418, 2331
\firstline 622	$\label{linegoal} $\lim \ensuremath{goal}_{\square}(\ensuremath{package}_{\square}\ensuremath{option}) \dots \underline{2328}$
\footnotetext	\linewidth 610,
\futurelet	644, 1126, 1415, 1416, 1417, 1937, 2331, 2445
576, 895, 952, 1683, 1703, 1723, 1790, 1853	\longtable 550, 2384
\FV@@CheckEnd	\longtabu
\FV@CheckEnd	\lowercase
\FV@DefineCheckEnd 1908, 1924, 2391	\LT@bchunk
\FV@EndScanning	\LT@cols
\FV@EnvironName 1920, 1921, 1923	\LT@echunk
\FV@LeftListFrame 1939	\LT@get@widths 1235

\LT@next 1752, 1753, 1763, 1774	\setbox 109, 123, 527,
\LT@p@ftntext	531, 648, 820, 1003, 1216, 1225, 1234,
\LT@startpbox 554	1284, 1541, 1601, 1608, 1654, 2249, 2256
\LTchunksize	\setlength 877, 2321
	\span 1784, 1785
M	\stretch 2156, 2158
\message 2342, 2352, 2356	\strutbox 110,
\minrowclearance	111, 115, 116, 126, 131, 691, 692, 1647, 1648
\multicolumn	
\multispan 844, 847, 1198, 1759, 1768, 1771	${f T}$
NT	\tabcolsep
N	\tabu 75, <u>539</u> , 1583
\narrowragged	\tabu@@celllalign
\NC@find 590, 797, 840, 853, 860,	677, 711, 1655, 1669, 2026, 2042, 2428
861, 865, 1017, 1020, 1029, 1033, 1068, 1523	\tabu@@celllalign@save 677, 711
\NC@rewrite@S	\tabu@@cellleft
\(\text{newcount}\) \(679, 713, 1657, 1671, 2024, 2039, 2430
\newdimen 30, 31, 32, 33, 38, 39, 40, 41, 42	\tabu@@cellleft@save 679, 713
\newsavebox 45, 46, 47, 48	\tabu@@cellralign
\newskip 2019	678, 712, 1656, 1670, 2027, 2043, 2429
\text{\text{newtabulinestyle}} \tag{16}, \frac{263}{263}, \frac{2442}{242}	\tabu@@cellralign@save 678, 712
\newtoks	\tabu@@cellright
\noalign	680, 714, 1658, 1672, 2025, 2040, 2431
189, 220, 247, 363, 367, 1238, 1245, 1652,	\tabu@@cellright@save 680, 714
1667, 1681, 1695, 1727, 1732, 1740, 1742,	\tabu@@DBG 122, 2345
1756, 1765, 1773, 1778, 1820, 1947, 2020	\tabu@@everycr 634, 1244
\noindent 1619, 1665, 1666	\tabu@@everypar 634
\numexpr 79, 162, 193, 1456, 1457, 1458, 1832, 2054	\tabu@@rowfontreset
O	$\dots \dots 682, 716, 2028, 2038, 2044, 2427$
\obeyspaces 69, 411, 1134	\tabu@@rowfontreset@save 682, 716
omit	\tabu@@spxiii
(OIIII 1720, 1749, 1709, 1704, 1709	\tabu@@tabudecimal 1871
P	\tabu@@thevline 1659, 1673
\PackageError 380, 549, 558, 867, 2016	\tabu@adl@fix 640, 2312, 2317, 2382
\PackageInfo	\tabu@afterendpar 606, 726, 731, 1051
\PackageWarning	\tabu@aftergroupcleanup 656,
151, 496, 935, 942, 2009, 2285, 2415	657, 659, 661, 1078, 1083, 1090, 1094, 1129
\parfillskip	\tabu@align 590,
\pdfadjustspacing	591, 1967, 1975, 1995, 1997, 2002, 2004
\pdfelapsedtime 1454, 2437, 2438	\tabu@aligndefault 587, 608,
\penalty 1756, 1765, 1773	610, 1508, 1967, 1975, 1978, 1997, 2004, 2008
\preamble 20, 988, 994, 1012, 1998	\tabu@aligndefault@text 1995, 2002, 2008
$\preamble_{\sqcup}(private_{\sqcup}column_{\sqcup}type) \dots 988$	\tabu@alloc <u>21</u> , 684, 722, 763
\prepnext@tok 635, 2078	\tabu@alloc@
\prevdepth 1625	\tabu@alloc@save
\protected@xdef 2203, 2212	\tabu@arc@G 322, 663, 664, 697, 698, 752, 2422
	\tabu@arc@Gsave
\mathbf{R}	\tabu@arc@L 315, 664, 747, 752, 759, 1519, 2372
\RaggedLeft 631, 925, 2144, 2146, 2242	\tabu@arith 1299, <u>1332</u> , 1405
\raggedleft 629, 921, 925, 2115, 2117, 2236	\tabu@arith@negcoef 1315, 1317, 1331
\RaggedRight 631, 923, 2132, 2134, 2243	\tabu@arithnegcoef <u>1314</u> , 1342, 1359, 1402
\raggedright . 629, 919, 923, 2103, 2105, 2237	\tabu@arstrut <u>109</u> , 2349
\rem@pt 1443, 1447, 1450	\tabu@arstrutbox
\remove@to@nnil 90	\tabu@AtBeginDocument 2365, 2366, 2413
\resetcolorseries 353, 365, 370	\tabu@begin
\rightskip 2126	\tabu@begincolor@b@x 2256, 2258
\romannumeral 1482, 1499, 1795	\tabu@beginfbox
\rowcolor 355, 367, 372, 1128	\tabu@box <u>45</u> , 1216, 1223, 1225, 1231,
\rowfont 18, 632, <u>2019</u>	1343, 1344, 1385, 1403, 1427, 1436, 1541,
Q	1556, 1558, 1559, 1563, 1564, 1570, 1601, 1608, 1613, 1614, 1615, 1626, 1627, 1633
S \save@decl	1608, 1613, 1614, 1615, 1626, 1627, 1633
\save@decl	\tabu@bufferX
\scantokens 59, 100, 412, 416, 1086, 1095, 1233	\tabu@C@extra 162, 182, 186, 190, 191
\scantokens_(package_option)	\tabu@c@l@r 83, 87, 90, 489, 490
	=

\tabu@C@linesep 193, 213, 217, 221, 222	\tabu@docline@evr
\tabu@c@loff 404, 420, 444, 489	\tabu@doclineafter 1829, 1830
\tabu@c@lon 404, 418, 419, 444, 490	\tabu@docolor@b@x 2255, 2263
\t abu@cell@align $\underline{2093}$, 2102 ,	\tabu@dp 1463,
2108, 2114, 2120, 2131, 2137, 2143, 2149	1466, 1472, 1474, 1618, 1630, 1644, 1648
\tabu@cell@C 2136, 2141, 2407	\tabu@dpdef 1618, 1630, 1638, 1643
\tabu@cell@c <u>2101</u> , 2407	\tabu@drsc@G 323, 665, 666, 699, 700, 753, 2423
\tabu@cell@J 2148, 2153, 2408	\tabu@drsc@Gsave 665, 699
\tabu@cell@j <u>2101</u> , 2408	\tabu@drsc@L 316, 666, 748, 753, 760, 1520, 2372
\tabu@cell@L 2130, 2135, 2405	\tabu@elapsedtime 726, 1082, 1093, 1102, 1197
\tabu@cell@l 2101, 2405	\tabu@end@envir 1160, 1590, 1591, 1594
\tabu@cell@R 2142, 2147, 2406	\tabu@endarray 640, 2268, 2273, 2280, 2410
\tabu@cell@r 2101, 2406	\tabu@endcolor@b@x 2259, 2260, 2264
\tabu@cellfont	\tabu@endenvir
\tabu@celllalign 161, 673, 707, 1655, 1660, 1669, 2026, 2042, 2086, 2089, 2094	1162, 1163, 1166, 1258, 1261, 1266, 1289 \tabu@endfbox 2251, 2252
\tabu@celllalign@def 161, 1106,	\tabu@endofcollect
1111, 1115, 1192, 1273, 1277, 2091, 2095	\tabu@endoftrials
\tabu@celllalign@save 673, 707	\tabu@endpbox
\tabu@cellleft	\t tabu@endpboxmeasure
675, 709, 1657, 1663, 1671, 1889, 2024,	\tabu@endquick
2031, 2039, 2087, <u>2089</u> , 2098, 2099, 2103,	\tabu@endrewrite 1028, <u>1032</u>
2109, 2115, 2121, 2132, 2138, 2144, 2150	\tabu@endrewritemulticolumn 834, 842
\tabu@cellleft@save	\tabu@endtabu 1590, 1597
\tabu@cellralign 674, 708, 1107,	\tabu@endtabus
1111, 1115, 1193, 1273, 1278, 1656, 1661,	\tabu@endtrial
1670, 2027, 2043, 2083, <u>2089</u> , 2096, 2097	\tabu@err 1292, 1293, 1294, 1313
\tabu@cellralign@save 674, 708	\tabu@everycr 1240, 1245
\tabu@cellright 158, 676,	\tabu@everyr@w 241, 244, 260
710, 1658, 1662, 1672, 2025, 2040, 2083, 2089	\tabu@everyrow
\tabu@cellright@save	\tabu@everyrow@bgroup 103, 234, 272, 283, 327
\tabu@cellrightfalse 2052, 2057, 2065	\tabu@everyrow@egroup <u>103</u> , 259, 281, 325, 377
\tabu@cellrighttrue	\tabu@everyrowfalse 56, 246, 1666
\tabu@cellskip 2019, 2157, 2170, 2171, 2172 \tabu@cellspacepatch 1547, 1552	\tabu@everyrowtrue 55, 696, 758, 845, 2077, 2436 \tabu@evr 261,
\tabii(0(:entering	262 1106 1111 1115 1192 1203 1272 1276
\tabu@Centering	262, 1106, 1111, 1115, 1192, 1203, 1272, 1276 \tabu@evr@G 257, 671, 672, 706, 729, 756, 2424
\tabu@centering 628, 2235	\tabu@evr@G 257, 671, 672, 706, 729, 756, 2424
,	\tabu@evr@G 257, 671, 672, 706, 729, 756, 2424
\tabu@centering	\tabu@evr@G 257, 671, 672, 706, 729, 756, 2424 \tabu@evr@Gsave 671, 706
\tabu@centering 628, 2235 \tabu@clarstrut	\tabu@evr@G 257, 671, 672, 706, 729, 756, 2424 \tabu@evr@Gsave
\tabu@centering	\tabu@evr@G 257, 671, 672, 706, 729, 756, 2424 \tabu@evr@Gsave
\tabu@centering	\tabu@evr@G 257, 671, 672, 706, 729, 756, 2424 \tabu@evr@Gsave
\tabu@centering	\tabu@evr@G 257, 671, 672, 706, 729, 756, 2424 \tabu@evr@Gsave
\tabu@centering	\tabu@evr@G 257, 671, 672, 706, 729, 756, 2424 \tabu@evr@Gsave
\tabu@centering	\tabu@evr@G 257, 671, 672, 706, 729, 756, 2424 \tabu@evr@Gsave
\tabu@centering	\tabu@evr@G 257, 671, 672, 706, 729, 756, 2424 \tabu@evr@Gsave
\tabu@centering	\tabu@evr@G 257, 671, 672, 706, 729, 756, 2424 \tabu@evr@Gsave
\tabu@centering	\tabu@evr@G 257, 671, 672, 706, 729, 756, 2424 \tabu@evr@Gsave
\tabu@centering	\tabu@evr@G 257, 671, 672, 706, 729, 756, 2424 \tabu@evr@Gsave
\tabu@centering	\tabu@evr@G 257, 671, 672, 706, 729, 756, 2424 \tabu@evr@Gsave
\tabu@centering	\tabu@evr@G 257, 671, 672, 706, 729, 756, 2424 \tabu@evr@Gsave
\tabu@centering	\tabu@evr@G 257, 671, 672, 706, 729, 756, 2424 \tabu@evr@Gsave
\tabu@centering	\tabu@evr@G 257, 671, 672, 706, 729, 756, 2424 \tabu@evr@Gsave
\tabu@centering	\tabu@evr@G 257, 671, 672, 706, 729, 756, 2424 \tabu@evr@Gsave
\tabu@centering 628, 2235 \tabu@clarstrut 690, 728 \tabu@clckpt 1139, 1218, 1227, 1280 \tabu@cleanup 654 \tabu@cline 1778, 1779, 1791 \tabu@clinearg 1781, 1792, 1807 \tabu@clinebox 1796, 1797, 1798, 1799, 1800, 1808 \tabu@clinebox 1796, 1797, 1798, 1799, 1800, 1808 \tabu@clineleads 1801, 1802, 1803, 1808, 1809 \tabu@clinescan 1702, 1707 \tabu@closetrialsgroup 1246 \tabu@cnt 21, 947, 949, 1001, 1023, 1028, 1122, 1265, 1271, 1340, 1341, 1358, 1378, 1459, 1460, 1489 \tabu@collectbody 598, 1135, 1156, 1587 \tabu@color@b@x 633, 2254, 2257 \tabu@color@b@x 633, 2254, 2257 \tabu@colortblalignments 2185, 2190, 2368 \tabu@colortblfalse 2369 \tabu@colortbltrue 2367	\tabu@evr@G 257, 671, 672, 706, 729, 756, 2424 \tabu@evr@Gsave
\tabu@centering	\tabu@evr@G 257, 671, 672, 706, 729, 756, 2424 \tabu@evr@Gsave
\tabu@centering	\tabu@evr@G 257, 671, 672, 706, 729, 756, 2424 \tabu@evr@Gsave
\tabu@centering	\tabu@evr@G 257, 671, 672, 706, 729, 756, 2424 \tabu@evr@Gsave
\tabu@centering	\tabu@evr@G 257, 671, 672, 706, 729, 756, 2424 \tabu@evr@Gsave
\tabu@centering 628, 2235 \tabu@clarstrut 690, 728 \tabu@clckpt 1139, 1218, 1227, 1280 \tabu@cleanup 654 \tabu@cline 1778, 1779, 1791 \tabu@clinearg 1781, 1792, 1807 \tabu@clinebox 1796, 1797, 1798, 1799, 1800, 1808 \tabu@clinebox 1796, 1797, 1798, 1799, 1800, 1808 \tabu@clineleads 1801, 1802, 1803, 1808, 1809 \tabu@clinescan 1702, 1707 \tabu@closetrialsgroup 1246 \tabu@cnt 21, 947, 949, 1001, 1023, 1028, 1122, 1265, 1271, 1340, 1341, 1358, 1378, 1459, 1460, 1489 \tabu@collectbody 598, 1135, 1156, 1587 \tabu@collectbody 598, 1135, 1156, 1587 \tabu@color@b@x 633, 2254, 2257 \tabu@color@b@x 633, 2254, 2257 \tabu@colortblalignments 2185, 2190, 2368 \tabu@colortblalignments 2369 \tabu@colortbltrue 2367 \tabu@commaXIII 411, 448 \tabu@DBG 125, 129, 1728, 1743, 2345, 2348 \tabu@DBG@arstrut 122, 2346 \tabu@debug 1560, 1616, 1628, 2352, 2353 \tabu@decimal 1849, 1903 \tabu@definestyle 399, 402 \tabu@DELTA 30, 1332,	\tabu@evr@G 257, 671, 672, 706, 729, 756, 2424 \tabu@evr@Gsave
\tabu@centering	\tabu@evr@G 257, 671, 672, 706, 729, 756, 2424 \tabu@evr@Gsave
\tabu@centering 628, 2235 \tabu@clarstrut 690, 728 \tabu@clckpt 1139, 1218, 1227, 1280 \tabu@cleanup 654 \tabu@cline 1778, 1779, 1791 \tabu@clinearg 1781, 1792, 1807 \tabu@clinebox 1796, 1797, 1798, 1799, 1800, 1808 \tabu@clinebox 1796, 1797, 1798, 1799, 1800, 1808 \tabu@clineleads 1801, 1802, 1803, 1808, 1809 \tabu@clinescan 1702, 1707 \tabu@closetrialsgroup 1246 \tabu@cnt 21, 947, 949, 1001, 1023, 1028, 1122, 1265, 1271, 1340, 1341, 1358, 1378, 1459, 1460, 1489 \tabu@collectbody 598, 1135, 1156, 1587 \tabu@collectbody 598, 1135, 1156, 1587 \tabu@color@b@x 633, 2254, 2257 \tabu@color@b@x 633, 2254, 2257 \tabu@colortblalignments 2185, 2190, 2368 \tabu@colortblalignments 2369 \tabu@colortbltrue 2367 \tabu@commaXIII 411, 448 \tabu@DBG 125, 129, 1728, 1743, 2345, 2348 \tabu@DBG@arstrut 122, 2346 \tabu@debug 1560, 1616, 1628, 2352, 2353 \tabu@decimal 1849, 1903 \tabu@definestyle 399, 402 \tabu@DELTA 30, 1332,	\tabu@evr@G 257, 671, 672, 706, 729, 756, 2424 \tabu@evr@Gsave

\tabu@get@decimal 1865, 1893	\tabu@leavevmodecolor
\tabu@get@decimalspace 1868, 1895	
\tabu@getc@l@r 483, 485, 495	\tabu@lefttok 2060, 2086, 2088
\tabu@getcolor 417, <u>481</u>	\tabu@linegoalt@rget 576, 577, 581
\tabu@getdecimal 1859, <u>1872</u>	\tabu@linegoaltarget 575, 576
\tabu@getdecimal@ 1850, 1853, 1859	\tabu@lines
\tabu@getdecimal@ignorespaces 1850, 1855, 1864	\tabu@lines@
\tabu@getline 266, 272, <u>388</u> , 804, 1804	\tabu@linesep
\tabu@getparam 406, 407, 463, 476, 480	\tabu@linestyle@
	· · · · · · · · · · · · · · · · · · ·
\tabu@gettarget 573, 574, 575	\tabu@longfalse 539
\tabu@Gextra 163, 188	\tabu@longpream
\tabu@givespace 1332, 1339, 1348	\tabu@longtrial
\tabu@Glinesep 194, 219	\tabu@longtrue
\tabu@gobblespace <u>60</u> , 568, 1066, 1861	\tabu@ls@ . 276, 279, 702, 761, 803, 813, 1793
$\verb \tabu@gobbletoken \underline{60}, 166,$	\tabu@ls@G 278, 279, 667, 668, 701, 702, 754, 2426
169, 171, 178, 180, 197, 200, 202, 209, 211	\tabu@ls@Gsave 667, 701
$\verb \tabu@gobbleX \dots \dots \dots \underline{60}, 1162, 1164$	\tabu@ls@L 275, 276, 668, 749, 754, 761, 1522
\tabu@gobblex 62, 63, 67	$\verb \tabu@LT@@hline 625, \underline{1751}$
\tabu@Grestore 190, 221, <u>224</u>	\t tabu@LT@startpbox
\tabu@Gsave 182, 186, 213, 217, <u>224</u>	\tabu@ltx@verb 1910, 1911
\tabu@halignto	\tabu@maybesiunitx <u>155,</u> 2094, 2377
573, 574, 584, 585, 873, 1048, 1267, 1280	\tabu@measuring 1194, 1209, <u>1212</u>
\tabu@header 1361, 1425, 1451, 1489	\tabu@measuringfalse 619, 1002, 1353, 1371, 1989
\tabu@hfil 2165, 2183, 2187	\tabu@measuringtrue 872, 1356, 1537
\tabu@hfill 2166, 2188	\tabu@message 878, 879, 1341, 1345, 1361, 1377,
\tabu@hfuzz 620, 1144, 1346, 1366, 1369, 1430, 1453	1390, 1392, 1397, 1455, 1526, 2356, 2360
\tabu@highprioritycolumns 792, 794, 1008	\tabu@message@arith 1345, <u>1425</u>
\tabu@hleads	$\t 0$ \tabu@message@defaulttarget 878, $\overline{1413}$
\tabu@hline 623, 1680, 1684	\tabu@message@endpboxmeasure 1560, 1575, 1580
\tabu@hlineAZ . 1676, 1677, 1678, 1679, <u>1695</u>	\tabu@message@etime 1197, <u>1454</u> , 2439
\tabu@hlineAZsurround 1697, 1699, 1705	\tabu@message@negcoef 1442
\tabu@hlinecorrection 1696, 1714	\tabu@message@reached 1377, 1451
\tabu@hlinescan 1701, 1706	\tabu@message@save 733, <u>1478</u> , 1983, 1986, 2357
\tabu@hsize 968, 972, 999, 1336, 1532, 1533, 1538	\tabu@message@spreadarith 1390, 1434
\tabu@hskip 2167, 2189	\tabu@message@target 879, 1423
\tabu@ht 1462, 1464, 1469, 1617, 1629, 1642, 1647	$\t 0$ \tabu@message@verticalsp $\underline{1461}$, $\underline{1616}$, $\underline{1628}$
\tabu@htdef 1617, 1629, 1638, 1641	\tabu@mkarstrut 119, 689, 720, 2346, 2349
\tabu@Hy@footnotetext 2214, 2219, 2232	\tabu@mkarstrut@save 689, 720
\tabu@Hy@ftntext <u>2199</u> , 2389	\tabu@mkpreambuffer 815, 842, 1002, 1038,
\tabu@Hy@ftntxt 2199, 2205, 2207, 2218	1044, 1060, 1078, 1090, 1120, 1130, 2076
\tabu@Hy@xfootnote <u>2199</u> , 2390	\tabu@modulo <u>60</u> , 352, 369
\tabu@if@nvir	\tabu@msg@align 1481, 1482, 1484, 1498
\tabu@ifcolorleavevmode 2036	\tabu@msgalign 1362, 1363, 1364, 1367,
\tabu@ifenvir	1426, 1427, 1428, 1431, 1435, 1436, 1437,
\tabu@ignorehfil	1438, 1439, 1465, 1467, 1470, 1475, 1481
2104, 2110, 2116, 2133, 2139, 2145, <u>2181</u>	\tabu@multicolumn 636, 843
\tabu@immediate	\tabu@multispan
\tabu@indent <u>603</u>	1665, 1784, 1785, 1831, 1832, 1835
$\verb \tabu@init$	\tabu@n@Gextra 164, 189
\tabu@justify 922, 926, 2121, 2123, 2125, 2128	\tabu@n@Glinesep 195, 220
\tabu@justifying 632, 2244	\tabu@naturalX
\tabu@keepls 808, 810, 817	\dots 30, 686, 717, 1203, 1563, 1565, 1566
\tabu@lasthline 621, <u>1676</u>	\tabu@naturalX@save 686, 717
\tabu@lasthlinecorrection 1678 , 1679 , 1738	\tabu@naturalXmax 688, 718, 1201,
\tabu@lastline	1202, 1383, 1385, 1437, 1565, 1566, 2433
\tabu@lastn@@p	\tabu@naturalXmax@save 688, 718
\tabu@lastnoop 2062, 2072, 2080	\tabu@naturalXmin 687,
\tabu@lastspace	719, 1202, 1384, 1385, 1438, 1567, 1568, 2432
\tabu@lasttry	\tabu@naturalXmin@save 687, 719
\tabu@latextwoe 2233, 2238, 2401	\tabu@nbcols <u>21</u> , 1238, 1507, 1665, 1836,
\tabu@LEADERS	1840, 1841, 1842, 1974, 2050, 2055, 2076
\tabu@leaders . 499, 500, 503, 819, 1810, 1813	\tabu@NC@list
\tabu@leaders@G 500, 522	613, 838, 1009, 1010, 1024, 1079, 1091
\tabu@leadersstyle	\tabu@negcoeffalse 544, 555
\tabu@leads 511, 518, 527, 531, 532, 535	\tabu@negcoeftrue 961, 1150

\tabu@nested <u>21</u> , 350, 353, 354, 355, 364, 365,	\tabu@rewritemulticolumn 833
366, 367, 370, 371, 372, 604, 722, 724,	\tabu@rewritevline 795, 796, 799
1039, 1040, 1050, 1099, 1185, 1186, 1270,	\tabu@rewriteX 852, 870, 984
1283, 1302, 1414, 1422, 1445, 1490, 1573,	\tabu@rewriteX@Ss 855, 857, 866
1574, 1641, 1642, 1643, 1644, 1996, 2003	\tabu@rewriteXrestore 882, 984
\tabu@nestedmeasure 583, 593, 601	\tabu@rewritten 797, 806, 807, 853, 865,
\tabu@new@columntype 769, 774, 777, 788	883, 888, 890, 942, 944, 960, 966, 968, 970
\tabu@newcolumntype <u>768</u> , 833, 995, 1019, 1022	\tabu@Rextra 188, 189, 190
\tabu@newlinestyle 264, 266, 271	\tabu@righttok 2058, 2066, 2081, 2085
$\verb \tabu@nexthlineAZ \underline{1695}$	\tabu@Rlinesep 219, 220, 221
\tabu@nocolor <u>153, 507, 1128, 2399</u>	\tabu@row@font
\tabu@noeverycr 1241, 1243	\tabu@rowc@lors 328, 329
\tabu@nohfil 2181, 2182, 2184, 2186	\tabu@rowcolors 327, 328, 336
\tabu@nolongtabu 558, 2385	\tabu@rowcolorseries 335, 337, 378
\tabu@normalcrbackslash 2246, 2248	\tabu@rowcolorserieserror 343, 380, 383
\tabu@norowcolor <u>153</u> , 1128	\tabu@rowfont 632, 2020
\tabu@nosiunitx 853, 856, 2378	$\t 0.00000000000000000000000000000000000$
\tabu@nowrite <u>142</u> , 1125	\t tabu@rowfontreset 248,
\tabu@noxfootnote 152	681, 683, 715, 2028, 2030, 2038, 2045, 2434
\t \tabu@noxfootnotes	\tabu@rowfontreset@save 681, 715
$\t 30, 403, 421, 423, 424, 425, 436, 489$	\tabu@rule@arc@ 308, 309, 326
\t tabu@offiii	\tabu@rule@drsc@ 290, 291, 295, 296, 307
\tabu@offxiii 468, 471, 477	\tabu@rulearc 287, 306, 308
$\verb \tabu@ofxiii \underline{446}$	\tabu@rulecolor 283, 284, 288
\tabu@on <u>30,</u> 403, 421, 422, 423, 425, 435	\tabu@ruledrsc 285, 289
\tabu@onxiii <u>446</u>	\tabu@ruledrsc@ 294, 295
\tabu@oXIII	\tabu@rulesstyle 437, 499, 1501
\tabu@oxiii	\tabu@sanitizearg 93, 267, 344, 345, 392, 1948
\tabu@pdftimer	\tabu@save
\tabu@pream	\tabu@save@decl
\tabu@prepnext@tok	. 1116, 1137, 1140, 1217, 1218, 1226, 1227
\tabu@prepnext@tokORI 635, 2070	\tabu@saved@ 751, 1958
\tabu@prev 1017, 1023, 1026	\tabu@savedecl 136, 138, 836, 1015
\tabu@printdecimal 1896, 1899, 1905	\tabu@savedparams
\tabu@private@columntype 779, 786, 789	\dots 733, 746, 757, 1504, 1965, 1968, 1976
\tabu@privatecolumns 787, 790, 837, 1016	\tabu@savedpream
\tabu@privatecolumntype	590, 1503, 1962, 1965, 1969, 1977
	\tabu@savedpreamble 1081, 1092, 1963
\tabu@pt 1446, 1448, 1479, 1480	\tabu@saveerr 981, 983, 989, 991, <u>2013</u>
\tabu@pushbegins 1157, 1177	\tabu@savels 816, 828, 839, 1000
\tabu@quick 1583, 1586, 1588, 1596	\tabu@savewarn 1949, 1950, <u>2013</u>
$\verb \tabu@quickend 1189, \underline{1287} $	\tabu@savewd 1960, 1961, 2012
\tabu@quickrule 598, 602, 1288	\tabu@scandecimal
\tabu@RaggedLeft 631, 2242	1851, 1853, 1854, 1861, 1866, 1869
\tabu@raggedleft	\tabu@select 1087, 1096, <u>1098</u>
\tabu@RaggedRight	\tabu@setcleanup 617, <u>654</u>
\tabu@raggedright 629, 2237 \tabu@raggedtwoe 2239, 2245, 2403	\tabu@seteverycr
\tabu@rc@ 249, 339, 340, 348,	1103, 1108, 1112, 1216, 1225, <u>1239</u> , 1282 \tabu@setextra 169, 171, 176, 178, 180
356, 361, 362, 368, 373, 374, 379, 704, 762	\tabu@setextrasep 173, 185, 187
\tabu@rc@G 340, 374, 669, 670, 703, 704, 755, 2425	\tabu@setlinesep
\tabu@rc@Gsave	\tabu@setreset
\tabu@rc@L 339, 362, 670, 755, 762	\tabu@sets@p
\tabu@rearstrut <u>109</u> , 253, 693, 1636, 1667	\tabu@sets@ve
\tabu@removehfil	\tabu@setsave 747, 748, 749, 766
2103, 2109, 2115, 2132, 2138, 2144, 2165	\tabu@setsep 200, 202, 207, 209, 211
\tabu@rescan <u>58</u> , 849, 1219, 1228, <u>1251</u>	\tabu@setstrategy 1117, <u>1121</u>
\tabu@reset 684, <u>732</u> , 830, 1239, 1636	\tabu@settarget 544, 555, <u>560</u>
\tabu@resetls	\tabu@setup 589, <u>603</u>
\tabu@restored 612, 1951, 1952	\tabu@shorttrial 1215, 1216, 1224
$\verb \tabu@restoreeverycr 1243, 1244 $	\tabu@siunitx 852, 854, 2378
$\verb \tabu@rewritefirst 591, 613, \underline{995}$	\tabu@siunitxerror 863, 867, 869
\tabu@rewritelast 1014, <u>1019</u>	\tabu@siunitxfalse
\tabu@rewritemiddle 1013, 1017, <u>1019</u>	\tabu@siunitxtrue 2376

\tabu@skipdecimal 1854, 1860, 1862, 1894	\tabu@usetabu 1054,
\tabu@spread 570, 574	1080, 1528, 1970, 1980, 1986, 1990, 1992
\tabu@spread@arith 1383, 1408, 1412	\tabu@verb 1907, <u>1910</u>
$\t 1298, \underline{1382}$	\tabu@verbatim 59, <u>1906</u>
\tabu@spreadfalse 544, 555, 1989	$\t 1272, \underline{1645}$
\tabu@spreadheader 1434, 1493	$\verb \tabu@verticalinit 1106, 1192, 1276, \underline{1635}$
\t \tabu@spreadtarget $\dots \dots \dots $ $\underline{30}$,	$\t 106, 1192, 1277, \underline{1599}$
874, 875, 1046, 1387, 1391, 1396, 1435, 1440	$\label{tabu@verticalsp@lcr} $$ \tabu@verticalsp@lcr 1602, $$ \underline{1612}$ $$$
\tabu@spreadtrue 574, 595	$\verb \tabu@verticalsp@pmb 1609, \underline{1621}$
\tabu@spxiii	\tabu@verticalspacing 1107, 1193, 1278, 1602, 1609
\tabu@start <u>21</u> , 235, 240, 331, 333, 334,	\tabu@vleads $\underline{45}$, 518
357, 1776, 1783, 1784, 1838, 1840, 1845, 1846	$\verb \tabu@vlinearg $
\tabu@start@stop 1836, 1837, 1847	\tabu@warn 151
\tabu@startpbox 614, 637	\tabu@warn@cellspace 2386, 2414, 2420
$\t 1204, 1206, 1531$	\tabu@warn@usetabu 1991, 2009
$\t 1207, 1549, \underline{1581}$	\tabu@warncolour 492, 496, 498
\tabu@startstop 1779, <u>1836</u>	\tabu@wd 974, 975, 1321, 1325, 1326,
\tabu@starttime 1456, 1457, 1458, 2438	1410, 1444, 1558, 1574, 1578, 1579, 2012
\tabu@starttimer 619, 2358, 2362	\tabu@wddef 1153, 1334,
\tabu@stop <u>21</u> , 235, 238, 1777, 1780,	1336, 1399, 1524, 1559, 1572, 1960, 2012
1785, 1839, 1840, 1841, 1842, 1843, 1846	\tabu@WRITE 145, 148, 150, 151
\tabu@stoptime 1454, 1456, 1457, 1458	\tabu@write
\tabu@strategy 1135, <u>1181</u> , 1311	\tabu@X
\tabu@strippt 964, 1479, 1482, 1499	\tabu@Xalign 919,
\tabu@strtrim	920, 921, 922, 923, 924, 925, 926, 934, 940
\tabu@tabarray 616, 639	\tabu@Xarg 870, <u>884</u>
\tabu@tabu@ 587, 589, 592	\tabu@Xc@ef 947, 949, 951 \tabu@xcline 1790, 1818, 1827
\tabu@tabudecimal 630, 1848, 1852, 1871	\tabu@Xcoef
\tabu@tabular 541, 546	\tabu@Xcoefs
\t tabu@target $\underline{30}$,	1047, 1049, 1142, 1315, 1348, 1383, 1960
561, 575, 584, 585, 594, 602, 605, 874,	\tabu@Xcol <u>21</u> , 885, 964, 1001,
876, 877, 1039, 1041, 1045, 1048, 1055,	1148, 1538, 1558, 1559, 1576, 1578, 1579
1080, 1141, 1144, 1151, 1185, 1264, 1288,	\tabu@Xd@sp 952, 953, 958
1308, 1321, 1326, 1344, 1347, 1360, 1364,	\tabu@Xdisp 886, 954, 967
1380, 1387, 1388, 1391, 1393, 1396, 1397,	\tabu@xfootnote 628, 1127, 2198, 2390
1421, 1424, 1428, 1439, 1446, 1450, 1505, 1539, 1540, 1542, 1570, 1972, 1991, 1993	\tabu@xfootnotetext 627, 2195
\tabu@textbar 100, 283, 835, 1004	\tabu@xhline 623, 1683, 1685, 1694
\tabu@thebody 43, 1083, 1084, 1086,	$\verb \tabu@xhlineAZ $
1094, 1095, 1159, 1221, 1230, 1253, 1257	\tabu@Xinit 1142, 1148, 1155
\tabu@thehleaders	\tabu@Xlcr 885, 935, 937, 938, 967
\tabu@thehline	\tabu@Xm@th 929, 952
\tabu@thehrule 504, 1811, 1815	\tabu@Xmath 886, 953, 967, 969
\tabu@theleaders 513, 520, 533, 537	\tabu@Xparse
\tabu@theparam	\tabu@Xparsespace
\tabu@therule 525, 528, 529, 530, 532, 534, 536	\tabu@Xrewritten 893, 959
\tabu@thestyle 270, 275, 278, 393, 395, 396, 398,	\tabu@Xsum . <u>38</u> , 1142, 1148, 1334, 1355, 1431 \tabu@Xtest 895, 897, 933
432, 433, 442, 803, 807, 808, 1781, 1793, 1810	\tabu@Xtype 916, 917, 918, 941, 945
\tabu@thetarget 612, 649, 877, 1416, 1417	\tabucline 16, 1700, 1709, 1713, 1776
\tabu@thevleaders 520, 821	\tabucline@scan
\tabu@thevline 806, 818, 827, 1659, 1664, 1673	\tabucolumn
\tabu@thevrule 504, 825	\tabucolX 30,
$\t 30, 403, 429, 430, $	888, 890, 949, 961, 962, 964, 974, 975,
434, 438, 439, 1706, 1707, 1719, 1720, 1721	977, 1040, 1141, 1142, 1145, 1152, 1186,
\tabu@titles 1341, 1490	1263, 1307, 1319, 1321, 1323, 1352, 1355,
\tabu@to 569, 573	1362, 1373, 1379, 1388, 1404, 1426, 1431,
\tabu@tr@cing 2339, 2341, 2364	1446, 1448, 1506, 1539, 1540, 1577, 1973
\tabu@tracing 2336, 2337, 2338, 2340	\tabucolX@init 1131, 1141, 1147, 1401
\tabu@tracing@save 2357, 2361	\tabudecimal $21,630, \underline{1848}$
\tabu@tracinglines 1788, 2342, 2343	\tabudefaulttarget 610,
\tabu@trial <u>1212</u>	612, 644, 650, 1126, 1415, 1418, 2331, 2445
\tabu@trialh@@k 140, 141, 1122	\tabuDisableCommands <u>140</u> , 379
\tabu@trimspaces 84, 87, 91	\tabuendlongtrial 1231, 1233, 1236
\tabu@trivlist 626, 2247	\tabular 546

\\ 1 1' m	0170
\tabulineoff 424, 2435	\unkern
\tabulineon 422, 2435	\unskip 139, 2175, 2308
\tabulinesep	\unvbox 517
\tabulinestyle $16, 272, 386, 2443$	\usetabu 19, 746, 980, 987, 1011, 1988, 2010, 2017
\tabuphantomline 12 , $\underline{1652}$	$\uberrule \uberrule \ube$
\tabureset	\$7
\taburow <u>21</u>	V
\taburowcolors 18, 249, <u>327</u> , 386, 2444	\value 2202, 2211
\taburulecolor 16, 283, 386	\vbox 517, 1607, 1798
	\vcenter 1606
\tabuscantokensfalse \ldots 1099, 2421	\vskip 518,
\tabuscantokenstrue 547, 557, 2333	1687, 1688, 1732, 1742, 1756, 1765, 1821
\tabustrutrule 39 , 122, 128,	\vss 518
132, 1283, 1616, 1628, 1731, 1747, 2347, 2350	
\tabuthepreamble 1086, 1095,	\vtop 820, 1531, 1570, 1605, 1799
1104, 1109, 1113, 1220, 1229, 1252, 1256	\mathbf{W}
\thempfn 2203, 2212	\write 143, 146, 1125
\thetaburow 21	
\tracingtabu 24, <u>2334</u>	\mathbf{X}
	$X_{\sqcup}(private_{\sqcup}column_{\sqcup}type)$ 852
${f U}$	\xglobal 354, 366, 371
$\verb \unhbox \dots \dots$	\xleaders 535, 1803, 1808